

**UNIVERSIDAD COMPLUTENSE DE MADRID**  
**FACULTAD DE GEOGRAFÍA E HISTORIA**  
Departamento de Geografía General Humana



TESIS DOCTORAL

**Diseño mediante ordenador en resolución de problemas de  
representación gráfica y cartográfica en geografía**

MEMORIA PARA OPTAR AL GRADO DE DOCTOR  
PRESENTADA POR

**Juan Antonio Cebrián de Miguel**

Madrid, 2015

TP  
1984  
192

Juan Antonio Cebrián de Miguel



x - 53-109104-9

DISEÑO MEDIANTE ORDENADOR EN RESOLUCION DE PROBLEMAS DE REPRESENTACION  
GRAFICA Y CARTOGRAFICA EN GEOGRAFIA

Departamento de Geografía General Humana  
Facultad de Geografía e Historia  
Universidad Complutense de Madrid  
1984



BIBLIOTECA

**Colección Tesis Doctorales. Nº 192/84**

© Juan Antonio Cebrián de Miguel  
Edita e imprime la Editorial de la Universidad  
Complutense de Madrid. Servicio de Reprografía  
Noviciado, 3 Madrid-8  
Madrid, 1984  
Xerox 9200 XB 480  
Depósito Legal: M-20392-1984

Juan Antonio Cebrían de Miguel

Diseño mediante ordenador en resolución  
de problemas de representación gráfica y  
cartográfica en Geografía.

Director: Cat. Dr. D. Joaquín Bosque Maurel

Departamento de Geografía General Humana.

Facultad de Geografía e Historia

Universidad Complutense de Madrid.

1.983.



#### AGRADECIMIENTOS.

Acabada ya la redacción del trabajo, le toca el turno a la tarea, ciertamente gustosa, de agradecer por escrito la colaboración de todos los que, por su actuación más o menos directa, han hecho posible los resultados que ahora se presentan. La relación ha de ser extensa, ya que han sido muchas las personas y circunstancias que se encuentran detrás de esta tesis.

En primer lugar quiero agradecer a D. Joaquín Bosque Maurel, Catedrático Director del Departamento de Geografía General Humana de la Facultad de Geografía e Historia de la U.C.M., la ayuda prestada en todo momento como director del trabajo. Su participación va más allá de lo estrictamente científico y técnico, ya que ha supuesto sobre todo un apoyo moral y una comprensión hacia los problemas planteados que me resulta imposible ahora valorar en sus justos términos.

También quiero manifestar decididamente que la participación de D. Miguel García Ferrández, profesor del Departamento de Ecuaciones Funcionales de la Facultad de Matemáticas de la U.C.M. y analista durante muchos años del Centro de Cálculo de esta misma Universidad, ha sido primordial en la conclusión de esta tesis. De su mano he aprendido la mayor parte de los requisitos técnicos -lenguajes de programación, fundamentos analíticos de la representación gráfica automatizada, conocimiento de sistemas operativos, etc.- que fundamentan el trabajo que presento. Sin su presencia, esta tesis no sólo no hubiera podido llevarse a cabo, sino ni siquiera plantearse como proyecto. Le estoy profundamente agradecido y espero seguir colaborando con él en sucesivas tareas de investigación.

A continuación, tengo que agradecer la colaboración del Instituto Geográfico Nacional, que ha posibilitado mi acceso al ordenador de su centro de proceso de datos. Este hecho ha sido decisivo en la viabilidad real de la presente tesis. Más concretamente, he de agradecer la colaboración de determinadas personas del Instituto, relacionadas directamente con el trabajo que he llevado a cabo durante este tiempo: D. Germán Vidal, Subdirector General de Investigación y Coordinación de Trabajos Geográficos; D. Antonio Gauchía, Jefe de Servicio del Centro de Estudios; D. Rafael Herrero, Jefe de la Sección de Sistemas de Información durante varios años y, actualmente, Jefe de la Sección de Geodesia Espacial; D. José María García-Courel, Jefe de la Sección de Cartografía Automática y D. Adolfo Izquierdo, Analista del Centro de Proceso de Datos.

También tengo que reconocer la ayuda prestada por la Cátedra de Matemáticas de la E.T.S.I. Industriales de Madrid, gracias a la cual pude realizar mi primer curso en lenguajes de programación -pro

## II

gramación en BASIC- en enero de 1980, y especialmente por su director D. José Alberto Jaén.

A D. Ignacio Sols, profesor adjunto de Algebra de la Facultad de Matemáticas de la U.C.M., le agradezco su ayuda en la definición de uno de los algoritmos de formación automática de envolventes de polígonos que presento en esta tesis.

D. José Ramón Colomines, ingeniero de imagen y sonido, ha sido la persona que ha realizado la filmación en video de la secuencia animada que representa la evolución temporal de las densidades provinciales españolas en el periodo 1900-1980.

En otro orden de cosas, ya que no se trata de colaboraciones directamente relacionadas con los trabajos que han constituido esta tesis, quiero manifestar los siguientes agradecimientos.

A los profesores Estébanez, Moreno Jiménez y Bosque Sendra, que me han precedido en el campo de los tratamientos numéricos en investigación geográfica, he de agradecerles, además de varias su gerencias, el hecho de haberme mostrado un camino a seguir. Al primero de ellos, además, he de agradecerle la propuesta de articular estos trabajos, cuando todavía se encontraban en fase de elaboración, y presentarlos como contenido fundamental de mi tesis doctoral en Geografía.

A los profesores Cole, Bradshaw y Mather, de la Universidad de Nottingham, agradezco vivamente las lecciones del I Curso de Técnicas de Cuantificación en Geografía, organizado por el I.G.N. y la Real Sociedad Geográfica de Madrid en diciembre de 1979, en el que también participaron el profesor Estébanez y el ingeniero geógrafo García-Courel. Este curso supuso el auténtico inicio de mi orientación hacia el estudio de problemas que poco a poco han ido decantando en los resultados que ahora se presentan.

También quiero aludir a todas las personas con las que he colaborado directamente en los últimos dos años, en proyectos de docencia e investigación en el campo de las aplicaciones informáticas en Geografía: Joaquín Bosque Sendra, Antonio Moreno Jiménez, Isabel Martínez de Sola, Vicente Rodríguez, Beatriz Jiménez, José Miguel Santos, Fermína Rojo, M<sup>a</sup> Jesús Vidal y Antonia Sáez. La existencia de este grupo de trabajo amplio y entusiasta, ha supuesto una ayuda fundamental en la elaboración de la presente tesis, ya que ésta, lejos de ser un trabajo aislado, ha sido considerada como un desarrollo más de las actividades de un equipo, alentado por todos sus componentes.

Mi agradecimiento también para todos los miembros del antiguo Departamento de Geografía Humana, con los que he convivido durante estos años de trabajo en la Facultad.

Finalmente he de agradecer la colaboración de mi padre, de D. José Moya, de D. David Pedroviejo y de D. Jesús Plaza, que me han prestado una ayuda inestimable en los trabajos de mecanografía, de lineación y confección definitiva del presente volumen.

ADVERTENCIAS.

El sistema adoptado para incluir las referencias bibliográficas, ha sido el de confeccionar una lista completa por orden alfabético que aparece al final del volumen. Los trabajos se citan por el nombre o nombres de su autor o autores seguidos del n° de orden, entre barras, de esa publicación en la lista general de referencias. Tres referencias bibliográficas no han sido incluidas en esta lista y se encuentran en una nota de la Introducción. La razón de esta excepción es que estas tres publicaciones han sido consultadas después de confeccionar la lista general; si se hubieran incluido en ésta, habría sido necesario reordenarla y modificar todas las citas del texto definitivo.

Se ha confeccionado un glosario de términos técnicos, que se encuentra a continuación del apéndice de subrutinas. La primera vez que aparece en el texto una de las palabras incluidas en él, se ha mecanografiado un asterisco.

Las figuras han sido numeradas por el capítulo al que pertenecen y por su orden dentro del capítulo.

Las expresiones algebraicas también han sido numeradas de forma semejante. Cada vez que se alude a alguna de ellas, se invoca al número que la identifica, entre barras.





IV

INDICE GENERAL

	<u>Págs.</u>
INTRODUCCION .....	1
I.1. TRATAMIENTO DIGITAL DE IMAGENES .....	3
I.2. ANTECEDENTES .....	4
I.3. APLICACIONES INFORMATICAS Y METODOLOGIA DE LA CIEN- CIA GEOGRAFICA .....	10
I.4. PROPUESTA DE UN SISTEMA INTEGRADO DE TRATAMIENTO DI- GITAL DE IMAGENES EN INVESTIGACION GEOGRAFICA .....	16
I.5. DESCRIPCION DEL CONTENIDO DE LA TESIS .....	21
I.6. ALGUNAS CONSIDERACIONES.....	24
CAPITULO 1. ELABORACION AUTOMATICA DE DIAGRAMAS.....	28
1.1. INTRODUCCION .....	29
1.2. TRATAMIENTO DIGITAL DE IMAGENES EN TECNOLOGIA VECTO- RIAL.....	29
1.3. DIBUJO AUTOMATICO DE PIRAMIDES DE POBLACION SIMPLES, COMPUESTAS O SUPERPUESTAS. EL PROGRAMA PIRAMID.....	36
1.4. DIBUJO AUTOMATICO DE DIAGRAMAS TRIANGULARES. EL PRO- GRAMA TRIAN. ....	75
CAPITULO 2. CARTOGRAFIA TEMATICA EN DOS DIMENSIONES: MA- PAS DE COROPLETAS .....	94
2.1. INTRODUCCION .....	95
2.2. DESCRIPCION DIGITAL DE AREAS EN TERMINOS VECTORIALES	96
2.3. RAYADO DE POLIGONOS .....	130
2.4. ELABORACION AUTOMATICA DE MAPAS DE COROPLETAS.....	138
CAPITULO 3. CARTOGRAFIA TEMATICA EN TRES DIMENSIONES: BLO- QUES DIAGRAMA EN PERSPECTIVA.....	159
3.1. INTRODUCCION .....	160

	<u>Págs.</u>
3.2. ESPACIO TRIDIMENSIONAL Y PROYECCIONES ISOMETRICAS..	161
3.3. RAYADO DE POLIGONOS EN TRES DIMENSIONES.....	168
3.4. MODELO DE REPRESENTACION DE VOLUMENES EN PERSPECTI- VA ISOMETRICA. EL PROBLEMA DE LA OCULTACION DE LI- NEAS. ....	174
3.5. DIVERSAS ALTERNATIVAS DE CONSTRUCCION DE UN VOLUMEN TEMATICO .....	177
CAPITULO 4. LA CUARTA DIMENSION: EL TIEMPO. ANIMACION	
GRAFICA DE BLOQUES DIAGRAMA .....	231
4.1. INTRODUCCION .....	232
4.2. ANIMACION GRAFICA Y SERIES HISTORICAS.....	233
4.3. EL PROGRAMA FILM .....	238
CONCLUSIONES .....	263
INDICE DE SUBROUTINAS .....	267
GLOSARIO .....	398
INDICE DE FIGURAS .....	402
BIBLIOGRAFIA .....	406

A

INTRODUCCION

	<u>Págs.</u>
I.1. TRATAMIENTO DIGITAL DE IMAGENES .....	3
I.2. ANTECEDENTES .....	4
I.2.1. Realizaciones previas en el ámbito de la Geografía Española .....	4
I.2.2. Estado de la cuestión en la Geografía Anglosajona y Centroeuropea .....	6
I.2.2.1. Aplicaciones informáticas al diseño de mapas y diagramas durante la década de los 60 .....	7
I.2.2.2. Rasgos fundamentales de la investigación en este campo en la década de los 70 .....	8
I.3. APLICACIONES INFORMATICAS Y METODOLOGIA DE LA CIENCIA GEOGRAFICA .....	10
I.3.1. Aspectos generales .....	10
I.3.2. Bases de datos espaciales .....	11
I.3.3. Representación gráfica automatizada y cartografía experimental .....	13
I.3.4. Algunos ejemplos .....	15
I.4. PROPUESTA DE UN SISTEMA INTEGRADO DE TRATAMIENTO DIGITAL DE IMAGENES EN INVESTIGACION GEOGRAFICA.....	16
I.4.1. La organización de la base de datos .....	16
I.4.2. Calidad de la representación y periféricos gráficos utilizables .....	17
I.4.3. Definición de tareas .....	19
I.5. DESCRIPCION DEL CONTENIDO DE LA TESIS .....	21
I.6. ALGUNAS CONSIDERACIONES .....	24

Resulta inevitable, en las primeras páginas de este trabajo, hacer una alusión -que va a ser breve- al tipo de concepción de la Ciencia Geográfica que lo ha hecho posible.

Tanto por el título, como por su contenido, las personas que se asomen a esta tesis no dudarán en calificarla de cuantitativa. Esta apreciación es, sin duda, cierta, pero este término ha sufrido en los últimos años un desgaste tal por su repetido uso, muchas veces sin un conocimiento real de su contenido, que creo necesario establecer una serie de puntualizaciones acerca de él, sin detenerme ahora en un examen exhaustivo del proceso de generalización de métodos cuantitativos en investigación geográfica, ya que esta tarea ha sido acometida recientemente por varios autores (1).

Partiendo de la lectura de estos textos y a raíz de una reflexión posterior creo interesante subrayar que, pese a la unidad y coherencia interna que se ha pretendido adjudicar a la llamada "Revolución Cuantitativa en Geografía", este proceso de cambio constituye más bien un conjunto, hasta cierto punto heterogéneo, de actitudes científicas y metodológicas.

En este sentido no me parece adecuada la interpretación del fenómeno como repercusión inmediata en nuestra ciencia del neopositivismo lógico, con todas las implicaciones de orden epistemológico e ideológico que esto lleva consigo. Por eso estoy totalmente de acuerdo con Bennett y Wrigley /11/ cuando afirman que la crítica actual al positivismo científico, aunque muy interesante como tal, es irrelevante desde el punto de vista de la geografía cuantitativa.

Es cierto que existe una vinculación entre las primeras figuras de la Geografía cuantitativa y los círculos en los que se fraguó el neopositivismo lógico y que, además, probablemente por su reducida extensión, los primeros grupos de trabajo en esta línea de investigación muestran un aspecto unificado, interesado por los problemas que se derivan de la aplicación del paradigma geométrico (vid. Bennett y Wrigley /11/) a los estudios de Geografía. Sin embargo, la evolución posterior de la corriente cuantitativa ha desembocado en la gran diversificación de métodos y estrategias que ahora contemplamos.

Por este motivo, nos parece mucho más lógico considerar el momento presente de la Geografía cuantitativa como un tipo de actitud ante el análisis de los fenómenos socioespaciales, que reúne todos, o algunos, de los siguientes elementos.

Preocupación por los problemas de tipo general y consideración de los casos reales como plasmaciones particulares de aquéllos.

Primacía de los modelos de razonamiento deductivo sobre los de carácter inductivo.

Insistencia en los aspectos de formalización del método científico y atención a los problemas de medida numérica de los fenómenos.

Colaboración con especialistas que desde el campo de las ciencias experimentales y de las matemáticas tienen intereses comunes con la investigación geográfica.

Valoración de los trabajos de prospectiva, por encima de las investigaciones de tipo histórico de la configuración actual de los fenómenos.

En estos términos, este trabajo puede ser considerado cuantitativo por cuanto pretende resolver, con carácter generalizado, los problemas de cartografía y representación gráfica de nuestra ciencia; porque en su desarrollo se ha llevado a cabo una formalización simplificadora de estas tareas y se ha abordado la descripción digital -numérica- de la configuración espacial, finalmente, porque es el resultado de una estrecha colaboración con matemáticos y especialistas en informática durante casi tres años.

Más concretamente, la tesis que introducen estas páginas, Diseño mediante ordenador en resolución de problemas de representación gráfica y cartográfica en Geografía, tiene por objeto iniciar un camino de recuperación del desfase de la Geografía española respecto de la Geografía occidental, fundamentalmente anglosajona y posteriormente centroeuropea, en cuanto a sus aportaciones en el área del tratamiento digital de imágenes.

### 1.1. TRATAMIENTO DIGITAL DE IMÁGENES.

Antes de seguir adelante, es importante aclarar el contenido del concepto "tratamiento digital de imágenes", a fin de evitar una posible confusión terminológica. Normalmente, se utiliza el término "tratamiento de imágenes digitales" -muy próximo al anterior- cuando se alude a las técnicas de captación y posterior manipulación de datos procedentes de los sensores remotos (scanners multiespectrales\*), que siempre constituyen una determinada imagen de la superficie o de la atmósfera terrestre, codificada y almacenada en forma numérica.

Sin duda, toda la tecnología de sensores remotos está incluida dentro del área del tratamiento digital de imágenes, pero no sólo ella. Los trabajos de cartografía automática, cartografía digital, reproducción de gráficos mediante terminales de ordenador, definición y puesta a punto de bases de datos espaciales, por citar algunos de los tópicos más habituales, forman parte, con el mismo derecho, de este campo de investigación.

El tratamiento digital de imágenes se configura, pues, como una técnica capaz de convertir en términos numéricos cualquier representación de la realidad espacial, para su almacenamiento en soportes\* electrónicos y posterior tratamiento por ordenadores, que sólo pueden asumir una representación simplificada -numérica- de dicha realidad.

La Geografía, que se define cada vez con más insistencia como ciencia espacial, no puede desentenderse del desarrollo de esta nueva tecnología, ni dejar de aportar sus particulares puntos de vista.

## I.2. ANTECEDENTES.

Si como hemos dicho unas líneas más arriba el objetivo de la presente tesis es paliar en parte un desfase de la Geografía española respecto de la Geografía norteamericana y europea en un campo determinado, es lógico que distingamos, a la hora de considerar sus antecedentes, las referencias bibliográficas previas en la literatura geográfica española -muy escasas-, de las que corresponden a publicaciones extranjeras -inabarcables en su totalidad y de las que presentamos tan sólo una muestra significativa-.

### I.2.1. Realizaciones previas en el ámbito de la Geografía española.

Todas las publicaciones anteriores al comienzo de los trabajos que han definido esta tesis se relacionan muy estrechamente con aplicaciones concretas del programa\* de cartografía automática SYMAP, desarrollado por el "Laboratory for Computer Graphics and Spatial Analysis" del "Harvard Graduate School of Design".

Este programa, importado a España por iniciativa del Instituto Geográfico Nacional y del Instituto de Geografía Aplicada del C.S.I.C. a principios de los 70, ha sido utilizado básicamente para solucionar problemas de cartografía temática de variables socioeconómicas, referidas a unidades espaciales administrativas, descritas por sus contornos.

Analizando las publicaciones españolas que hacen referencia bien a las prestaciones del programa, bien a aplicaciones concretas del mismo, podemos distinguir un primer grupo de trabajos, que se alinean en la primera de las direcciones apuntadas, llevados a cabo por técnicos del Instituto Geográfico Nacional, algunos de los cuales han visto la luz en revistas científicas de carácter geográfico o de planificación territorial. Nos referimos a las aportaciones de Núñez de las Cuevas /126/ y García-Courel /61, 62 y 63/.

También pertenece a este momento inicial la traducción de un artículo de Gachter y Kilchemann /58/ sobre el tema, publicado en la revista Geographica del Instituto de Geografía Aplicada del C.S.I.C.

A partir de 1975 se inicia la difusión del empleo de este tipo de procedimientos\* de cartografía automática en los medios universitarios de la Geografía española. La aplicación de esta nueva tecnología aparece centrada -al menos por lo que respecta a la producción bibliográfica concreta- en dos focos: Zaragoza y Madrid. En los trabajos de Faus Pujol y Calvo Palacios /54 y 55/ y de Moreno Jiménez /121 y 122/ se utiliza este programa de cartografía en análisis de estructura urbana espacial. En el trabajo de Bosque Sendra /22/ se confía al programa SYMAP la cartografía de variables electorales sobre una base provincial y municipal.

Reflejo de la creciente atención que se va despertando en la Geografía española hacia la utilización de programas de cartografía automática son, también, el artículo de Chapot Blanquet /37/ en la Revista de Geografía de la Universidad de Barcelona y una reseña de Díaz Álvarez /47/ a un libro sobre el tema aparecida en la revista Paralelo 37°.

En este ambiente de interés por la aplicación de técnicas informáticas al proceso de representación gráfica y cartográfica en Geografía, se formula la presente tesis, como un intento de ampliación sustancial de los procedimientos automáticos de representación utilizados hasta ese momento por un sector de la investigación geográfica española. A este planteamiento contribuye en no menor medida la organización, en diciembre de 1979, del I Curso de Técnicas de Cuantificación en Geografía, patrocinado por el Instituto Geográfico Nacional y la Real Sociedad Geográfica de Madrid.

Con motivo de la celebración de este curso, impartido por los profesores ingleses de la Universidad de Nottingham Cole, Bradshaw y Mather, el profesor Estébanez de la Universidad Complutense de Madrid y el ingeniero geógrafo García-Courel, jefe de la Sección de Cartografía Automática del Instituto Geográfico Nacional, se inicia un progresivo acercamiento de varios miembros del Departamento de Geografía Humana de la Facultad de Geografía e Historia de la Universidad Complutense de Madrid a problemas que están presentes en el marco de la investigación geográfica anglosajona actual, algunos de los cuales están relacionados con las actividades que venían desarrollando, sin conocimiento real de la Geografía universitaria española, las Secciones de Sistemas de Información y de Cartografía Automática del Instituto Geográfico Nacional, desde hacía unos años.

Resultado de este contacto más directo con la tecnología de aplicaciones informáticas a trabajos de Geografía es la constitución de un grupo de investigación, centrado en torno al Departamento de Geografía Humana de la Universidad Complutense de Madrid, que aspira a su reconocimiento institucional por parte de las au



toridades académicas como Laboratorio de Análisis Geográfico y Cartografía Automática (L.A.G.C.A.). Las actividades del grupo se reparten entre la docencia y la investigación. Dentro de esta última línea, y atendiendo solamente a la vertiente de aplicaciones informáticas a tareas de representación gráfica y cartográfica, se formulan diversos objetivos, algunos de los cuales ya se han alcanzado, habiéndose presentado sus resultados bien en forma de Memoria de Licenciatura, o, como en el presente caso, en forma de Tesis Doctoral, bien en forma de publicaciones en revistas de Geografía o de comunicaciones a congresos científicos.

Como muestra de trabajos relacionados con la utilización de procedimientos\* de cartografía automática ya desarrollados (SI MAP), se pueden citar las aportaciones de Bosque Sendra /23/, Fernández García y Moreno Jiménez /56/, Moreno Jiménez /123/, Requés Velasco /133/ y Sáez Cala /147/.

En la línea del desarrollo y aplicación de nuevos sistemas de representación hay que destacar, por su novedad -es la primera vez que se presenta en el ámbito de la Geografía española un trabajo que incluye cartografía temática de variables socioeconómicas mediante bloques diagrama en perspectiva isométrica-, la comunicación de Bosque Maurel, Cebrián de Miguel y Bosque Sendra /19/ al VII Coloquio de Geografía de Pamplona. En la misma dirección se encuentran los trabajos de Bosque Sendra /23/, Martínez de Sola /105/, Rojo Pérez /143/, Bosque Maurel /20/, Cebrián de Miguel y Bosque Sendra /34/, Cebrián de Miguel y García Ferrández /35/, Requés Velasco /133/ y Rojo Pérez /144/.

Plasmación de la colaboración, durante este tiempo, con miembros del Instituto Geográfico Nacional -siempre ciñendonos exclusivamente al campo de la representación gráfica y cartográfica-, es la formulación del proyecto de investigación Sistema de tratamiento automático de la información espacial para la planificación física y la ordenación del territorio presentado ante la Comisión Asesora de Investigación Científica y Técnica, en el que participan, junto a miembros del Instituto Geográfico Nacional y profesores de la Universidad Politécnica de Madrid, dos miembros del Departamento de Geografía Humana de la Universidad Complutense de Madrid. También en esta línea se encuentra la comunicación de Herrero García y Cebrián de Miguel /77/ al "Urban Data Management 9th European Symposium".

Finalmente, también se puede reseñar la colaboración, en la utilización de nuevas técnicas de representación para la ilustración de resultados, en algunos libros publicados recientemente (Capel y Urteaga /31/ y García Ballesteros /60/).

### 1.2.2. Estado de la Cuestión en la Geografía anglosajona y centro europea.

Contrastando claramente con la escasa producción bibliográfica de la geografía española en el campo de la automatización de

la representación gráfica y cartográfica, las referencias sobre este tópico en las publicaciones norteamericanas y centroeuropeas son de tal magnitud, que resulta muy difícil estructurarlas con vistas a lograr una comprensión esquemática de las líneas fundamentales de innovación en este campo. En esta tarea, la consulta de los trabajos de Dawson y Unwin /45/, McDougall /97/ y Bennett y Wrigley /11/ ha sido decisiva (2).

#### 1.2.2.1. Aplicaciones informáticas al diseño de mapas y diagramas durante la década de los 60.

Las primeras noticias de la aplicación de procedimientos informáticos a la resolución de problemas espaciales -incluido el de la representación gráfica y cartográfica- proceden de diversas universidades norteamericanas y se producen en los primeros años de la década de los 60. Destacan especialmente las aportaciones llevadas a cabo en las universidades de Washington, Chicago, Michigan y Northwestern.

El ritmo de trabajo en este campo es tal, que cinco años después se puede decir que la nueva tecnología ha sido asumida en grado considerable por casi todos los departamentos de Geografía del país. Los cursos de programación se consideran obligatorios para todos los alumnos que pretendan obtener la licenciatura en Geografía o en Planificación Territorial. Se organiza un servicio de intercambio de programas de ordenador entre entidades docentes o de investigación de carácter internacional: "Geography Program Exchange" (3). En muchos departamentos se dispone de una biblioteca de programas permanentemente actualizada, que cubre los siguientes campos: cartografía, análisis estadístico-espacial y problemas de optimización.

Al final de los años 60 las aportaciones individuales van adquiriendo formulaciones globales y coherentes; esto propicia la constitución de entidades específicas en las universidades, dedicadas al desarrollo de programas de uso general -tal es el caso del "Laboratory for Computer Graphics & Spatial Analysis" del "Harvard Graduate School of Design"- . Con la aparición de estas nuevas instituciones se potencia la difusión de esta tecnología en el exterior de los EEUU.

Analizando la producción bibliográfica correspondiente a este periodo, con vistas a destacar sus rasgos más característicos, se pueden establecer las siguientes puntualizaciones.

Desde el punto de vista cronológico hay que señalar que la primera referencia que hemos podido descubrir se remonta al año 1959. En ella se aborda el tema de la automatización de la cartografía (Tobler /165/ -profesor de Geografía de la Universidad de Michigan-). Así, el problema fundamental que pretende resolver la presente tesis dentro del marco de la Geografía española está ya planteado en los departamentos de Geografía de algunas universidades norteamericanas veintitrés años antes.

Por lo que respecta al contenido de las publicaciones, la atención se vuelca preferentemente hacia tres problemas básicos: cartografía automática mediante coropletas e isopletas, animación gráfica y definición de bases de datos espaciales. Junto a las referencias ya aludidas, se pueden citar, por ejemplo, los trabajos de Jenks y Knos /83/, Robinson /141/, Kao /88/, Knoulton /93/, Monnier /116, 117 y 118/, Tobler /167, 168 y 169/, Cornwell y Robinson /40/, Wilson /180/, Marble /102/, Gaits /59/, Kern y Rushton /89/, Rohlf /142/, Crain /42/, Horwood /78/ y Murray /124/.

Otro rasgo importante de las publicaciones sobre este tema es la colaboración de profesionales ajenos a nuestra disciplina en la definición de modelos y sistemas de representación. Junto a publicaciones aisladas procedentes del campo de la meteorología (Sawyer /150/), de la física (Heap y Pink /74/), o de las matemáticas aplicadas (Garfinkel /64/, Shephard /153 y 154/ y Thayer y Storer /161/), destaca la aportación de geólogos y geofísicos, miembros del "Kansas Geological Survey" en su mayor parte (Miller /110/, Merriam y Sneath /107/, Merriam y Cocke /108/ y McIntyre, Pollard y Smith /98/).

En cuanto a las publicaciones periódicas de mayor número de referencias, destacan las revistas Annals of the Association of American Geographers, Professional Geographer y Cartographic Journal. La presencia insistente de artículos sobre el tema en esta última revista es una muestra de la estrecha vinculación que existe en EEUU, a diferencia de lo que ocurre en el caso español, entre la Cartografía científica y la Geografía.

Antes de pasar a analizar la situación en la década siguiente, caracterizada por la difusión masiva de estas técnicas por toda Europa occidental, hay que subrayar la existencia de ciertos estudios pioneros en Europa en los últimos años 60. La figura clave es David P. Bickmore que, desde la "Experimental Cartography - Unit, Royal College of Art" de Oxford, inicia el camino de la cartografía digital en Europa, a través de sus publicaciones (Bickmore /14, 15 y 16/) y de su actividad docente dirigida a alumnos de diversos países europeos. Otros trabajos de esta etapa inicial son, por ejemplo, los de Rosing /145/, Sentance /152/, Dyer y Wellman /52/ y Tomlinson /170/.

#### 1.2.2.2. Rasgos fundamentales de la investigación en este campo en la década de los 70.

Con el cambio de década, los geógrafos británicos se incorporan plenamente al proceso de definición de nuevas aplicaciones informáticas a la investigación geográfica, y, más concretamente, por lo que a esta tesis afecta, a las tareas de representación gráfica y cartográfica.

Junto a la proliferación de publicaciones sobre el tema en los medios geográficos británicos, dos testimonios avalan la realidad de lo que acabamos de afirmar en el párrafo anterior. En pri

mer lugar disponemos de la crónica de Bennett y Wrigley /11/ sobre los principales temas discutidos en las conferencias anuales del QMSG ("Institute of British Geographers Quantitative Methods Study Group") desde su creación en 1964. Allí podemos comprobar la aparición en escena en el periodo 1972-75 -prolongándose ya hasta el presente- de ponencias y comunicaciones acerca de temas tales como cartografía automática, técnicas de interpolación espacial y sistemas de información geográfica.

El otro punto de referencia lo constituye la celebración en la universidad de Nottingham, en julio de 1973, de la reunión científica "Nato Advanced Study Institute. Display and Analysis of Spatial Data". En esta conferencia, patrocinada en colaboración por el "Kansas Geological Survey, University of Kansas" y el "Department of Geography, University of Nottingham", se hizo manifiesto, tanto en el número de participantes como en el número y calidad de las comunicaciones presentadas, el elevado nivel alcanzado por la Geografía y Cartografía británicas -no tan disociadas entre sí como en el caso español- en el campo de la representación gráfica y análisis espacial mediante ordenador. (cfr. Davis y McCullagh /44/).

También se produce en esta década la incorporación francesa, un poco más tardía y con otras características, a este campo de la investigación. Las primeras referencias que hemos podido descubrir son del año 1972. Por otra parte, se trata fundamentalmente de trabajos de matemática aplicada o de aportaciones del "Service des Applications Nouvelles" del I.G.N. francés. Al igual que ocurre en España, la Cartografía y la Geografía francesas no tienen, por el momento, excesivos puntos de contacto y colaboración. La única excepción a esta norma la constituye la presencia en este campo de los geógrafos miembros del "Laboratoire de Cartographie Thématique" de la Universidad Louis Pasteur de Estrasburgo (cfr. Rimbert /140/).

Mucho más esporádicas, aunque esto puede ser debido, como ya hemos señalado, a la procedencia de las fuentes bibliográficas a las que hemos tenido acceso, nos parecen las aportaciones de otras comunidades europeas: alemanes (Boyle y Kröll /26/), holandeses (Tuerke /174/), soviéticos (Frolov /57/) y suizos (Grandjean y Arnold /68/).

Por lo que respecta al contenido de los trabajos publicados en la década de los 70, se aprecia en primer lugar la presencia de estudios sobre temas que ya habían sido tenidos en cuenta en la década anterior: cartografía automática mediante isopletas y coropletas (cfr. Howarth /79/, Kadmon /86 y 87/, Rhind /134/, Jarō /81/, Peucker /131/, Baxter /8/, Baker /5/, Olea /127/, Jeffery, O'Hare y Board /82/, Mallet /101/, Virginia Polytechnic Institute and State University /177/ y Utano /175/), cartografía dinámica mediante ordenador (cfr. Anderson /2/, Moellering /111, 112, 113, 114 y 115/ y Hackathorn /71/) y definición de bases de datos espaciales (cfr. Petit /130/, Tomlinson /171/, Merrill /109/, Bouillé

/24 y 25/, Peucker y Chrisman /132/, Kirby y Tarn /90/, Rhind /136, 137 y 138/, Douglas /51/, Klinger /92/, Haralick /73/ y Cox, Aldred y Rhind /41/).

Junto a ellos, aparecen nuevos problemas que antes no habían sido tenidos en cuenta. Es en esta década cuando se considera la aplicación de la representación automática de volúmenes en perspectiva a la cartografía de variables georeferenciadas (cfr. Douglas /49/, Sprunt /157 y 158/, Crawford y Marks /43/, Mallet /1017, Rowles /146/, Monmonier /119/, Brodlie /28/ y Taylor y Fraser /160/). También se plantea en estos años la solución automatizada de cuestiones muy técnicas, tales como la edición de información espacial y la generalización cartográfica (cfr. Brophy /29/, Douglas y Peucker /50/, Maling /100/, Rhind /135/, Marino /103/ y Jenks /84/). Otros nuevos tópicos son la definición de sistemas de cartografía interactiva (cfr. Newman y Sproull /125/ y Bell y Bickmore /10/), técnicas de interpolación espacial (cfr. Macomber /99/, Basset /7/, Koelling y Whitten /94 y 95/ y Campbell /30/) y la aplicación de la representación gráfica al análisis estadístico numérico (cfr. Jones /85/, Anscombe /4/, Gnanadesikan /67/, Everitt /53/ y Guermond /70/).

También aparecen por primera vez, en estos años, manuales escritos por geógrafos sobre aplicaciones informáticas a los diversos campos de investigación y estudio en nuestra ciencia (cfr. McDougall /97/, Dawson y Unwin /45/ y Mather /106/).

Finalmente, por su importancia en la justificación de esta tesis, hemos podido reseñar diversas tesis doctorales y masters leídos durante los años 70 en universidades anglosajonas sobre temas muy próximos a los que han configurado ésta. Nos referimos a los trabajos de Tomlinson /172/, de la Universidad de Londres, Waugh /178/ y Thomas /162/, de la Universidad de Edimburgo, Traylor /173/, de la Universidad de Kansas y Adams /1/, de la Universidad de Durham.

### I.3. APLICACIONES INFORMATICAS Y METODOLOGIA DE LA CIENCIA GEOGRAFICA.

#### I.3.1. Aspectos generales.

Si bien, en una primera evaluación del contenido de la presente tesis, se puede llegar a la conclusión, que es en parte cierta, de que se trata de un trabajo básicamente instrumental, en el sentido de que tan sólo se limita a ofrecer a la investigación geográfica unas herramientas de diseño gráfico y análisis espacial de potencia muy superior a la de las que hasta el momento son normales en los trabajos realizados en nuestro país, no puede dejarse de lado en ningún momento la consideración de que, en todos los casos,

la aparición de nuevas capacidades de análisis ha provocado una ampliación de objetivos metodológicos y conceptuales en todas las ciencias.

Más concretamente, la asimilación por parte de la Geografía de la tecnología informática -y esta tesis supone avanzar elementos en pro de esta transformación, en el campo concreto de las tareas de representación gráfica y cartográfica-, está provocando cambios importantes en los planteamientos básicos de nuestra ciencia.

La utilización del ordenador permite, como ventaja fundamental, el tratamiento de volúmenes de datos descomunales a velocidades de vértigo, pero requiere que tanto el tipo de datos a manipular como las operaciones a realizar con ellos sean sumamente sencillos.

Una consecuencia inmediata, que se deriva directamente de este hecho, es la posibilidad de considerar un número de variables explicativas, de cualquier configuración de los hechos espaciales, de orden incomparablemente superior al habitual en los trabajos de Geografía tradicionales. El volumen de datos de base que fundamenta cualquier investigación geográfica que utiliza medios informáticos es cada día mayor, y, coherentemente con ello, las conclusiones que se derivan, más ampliamente generalizables.

Ahora bien, es un hecho, como acabamos de comentar, que los datos que es capaz de manipular un ordenador han de tener una estructura muy simple -numérica, en la mayoría de los casos-. Este hecho exige, por tanto, una traducción en términos digitales de la realidad geográfica, no solamente de los valores de las características que afectan a distintos puntos del espacio sino también de la propia configuración espacial. Esta trasposición de los términos lleva sin duda a una visión esquemática de la realidad, ya que se trata de una simplificación de la misma, pero, precisamente por eso, permite la generalización de procesos y situaciones al hacer más fácilmente comparables sus diversos estados.

La utilización del ordenador en Geografía, consolida la posición de los estudios teóricos -de modelos y sistemas-, haciéndolos realmente viables, y propicia una mayor formalización y una estructuración más rigurosa y generalizadora de nuestra ciencia.

### I.3.2. Bases de datos espaciales.

Otra consecuencia que se deriva de la posibilidad de tratamiento de grandes conjuntos de datos, es la preocupación cada día mayor, por los trabajos de documentación geográfica. Este hecho ha dado lugar a la aparición en escena de nuevo tópico, con personalidad propia, en la literatura geográfica: diseño e implementación de bases de datos espaciales -muy próxima al término de geocodificación-. Las realizaciones de la Geografía española en este cam-

po son prácticamente inexistentes, (4) pero no resulta aventurado asegurar que será muy próximamente tomado en consideración, ya que, en las circunstancias actuales, permanecer al margen de esta línea de trabajo supone la condena definitiva a trabajos improbos de búsqueda de documentación, y la renuncia, de una vez por todas, a rebasar la frontera de la subactualidad en los trabajos de análisis geográfico.

En pocas palabras, el diseño de una base de datos espaciales supone el almacenamiento topológicamente estructurado en un soporte informático, de la configuración básica y de la articulación del terreno en consideración, así como los valores de las características que afectan a sus diversos elementos, con vistas a obtener una rápida respuesta ante cualquier demanda de información y a facilitar una continua actualización de los datos.

Ya existe en España un modelo de base de datos espaciales, desarrollado por el IGN, pero se encuentra en una etapa embrionaria por falta de contenido. La colaboración de los departamentos de Geografía universitarios en este proyecto puede ser decisiva para su revitalización y los beneficios mutuos, considerables.

Hasta el momento nos hemos detenido en la consideración de la necesaria simplificación de los datos a introducir en el ordenador, para que éste pueda eficazmente asimilarlos. Llega el momento de referirnos a las operaciones de manipulación de los datos que van más allá de su almacenamiento y búsqueda indiscriminada en una memoria \* digital. Lógicamente, cuanto mayor es el volumen de datos que se considera, menor significado tiene la consideración de los datos aisladamente. Todo el esfuerzo de almacenar electrónicamente una ingente información resultaría inútil si no se dispusiera de procedimientos de síntesis que describan con facilidad las características más notorias que se puedan extraer de la consideración simultánea de un elevado número de variables explicativas.

El desarrollo de la ciencia estadística ha propiciado modelos de síntesis de la información, cuya formulación es, en ocasiones, enormemente complicada. Por este motivo ha sido necesario acometer un trabajo de descomposición de las operaciones de manipulación y cálculo más complejas en sus elementos más simples -operaciones aritméticas básicas-, que son las únicas que realmente pueden llevar a cabo un ordenador. Estas tareas de programación se han llevado a cabo normalmente fuera del ámbito de la investigación geográfica, aunque, no es el caso de España, existen geógrafos, con importantes conocimientos estadísticos, que están trabajando en este campo con vistas a introducir en los modelos estadísticos convencionales las peculiaridades del tratamiento de datos espaciales.

Finalmente, a partir de las pautas de comportamiento espacial más firmes, detectadas por procedimientos de análisis estadístico, se formulan modelos de simulación que intentan, en fun -

ción de diversas hipótesis, la plasmación de escenarios futuros. Estos modelos, en su versión informática -única realmente factible-, convierten al ordenador en un auténtico laboratorio experimental de hipótesis de evolución futura.

### I.3.3. Representación gráfica automatizada y cartográfica experimental.

Paralelamente a todos los procedimientos que acabamos de comentar, existe una necesidad básica a cubrir, para no perder el punto de vista y la significación real de los hechos, que traducidos y manipulados digitalmente no son directamente interpretables. Me estoy refiriendo a la síntesis gráfica de la información almacenada numéricamente en un ordenador, bien de la información primaria -datos de base-, bien de los resultados de la manipulación de ésta. Esta síntesis, que es muy necesaria en el caso de datos espaciales -diagramas-, es absolutamente imprescindible cuando los datos en consideración tienen una componente espacial -mapas-.

A esta tarea, que como hemos señalado más arriba ha sido acometida ampliamente por la Geografía anglosajona y centroeuropea, pero que no tiene más que algunos precedentes en la Geografía española, hemos dedicado el contenido de la presente tesis doctoral.

El tema no ha sido agotado, ni mucho menos, en su totalidad y, por tanto, todavía serán necesarios unos cuantos años de trabajo sobre esta materia para saldar definitivamente, al menos a grandes rasgos y en sus puntos críticos, el déficit acumulado por la Geografía española. A pesar de ello, los resultados que se presentan en esta tesis son ya de una cierta importancia y, sobre todo, constituyen un punto de partida firme sobre el que levantar nuevas realizaciones, algunas de las cuales se encuentran ya en fase avanzada de elaboración.

Todo trabajo humano está compuesto por dos fases claramente diferenciadas: tareas mecánicas y repetitivas, y tomas de decisión y formulaciones realmente innovadoras. Pues bien, la primera y más inmediata racionalización de la representación gráfica consiste en poder liberar al investigador de todo aquello en lo que él no está aportando realmente algo creativo e innovador. Tradicionalmente esto se ha solventado más o menos acertadamente recurriendo a la división del trabajo y encomendando las tareas de diseño puramente mecánico a una persona cuyo trabajo consiste precisamente en eso; ahora bien, si se asimila la tecnología de tratamiento digital de imágenes, el resultado será cualitativamente distinto.

En una Memoria de Licenciatura, leída recientemente en esta Facultad (Martínez Sola /105/), se exponen detenidamente las ventajas de la utilización de esta tecnología, que, por supuesto, no es viable sin la asistencia del ordenador. Tres de ellas me parecen especialmente dignas de consideración: economía, rapidez y pre-



cisión.

La combinación de estas tres características de la representación gráfica automatizada posibilita la incorporación de la misma al proceso científico de toma de decisiones y de verificación o rechazo de hipótesis de investigación.

Hasta el momento presente, la representación gráfica es considerada producto final en la Geografía española, utilizándose para sintetizar y presentar a los lectores los resultados y conclusiones de la investigación. Normalmente la serie estadística (cuadro), cuya información sintetiza, también se publica para satisfacer los deseos de una mayor precisión numérica y complementar el carácter generalizador y, por tanto, nivelador del mapa temático o diagrama. Esto no podía ser de otra manera, dado el elevado costo en dinero, o en dedicación personal del investigador, de la tarea de diseño.

La rapidez y economía de las técnicas de tratamiento digital de imágenes posibilitan la realización de continuos ensayos, a lo largo de la investigación, de representación gráfica de las características intrínsecas de las variables estudiadas y de las distribuciones espaciales de los fenómenos que se analizan, que constituyen un valioso banco de pruebas de las hipótesis de trabajo. Ha llegado por tanto el momento de que entre el material de trabajo propio del investigador proliferen los borradores, diagramas, mapas de distribución espacial de variables, etc, provisionales, que no interesan en sí mismos, como resultado, sino como plasmaciones temporales y contingentes de cierto estado de los trabajos de análisis de determinados hechos de carácter geográfico.

En Geografía partimos siempre de una imagen real que intentamos conceptualizar y definir -cada vez es esto más frecuente- en términos numéricos, para su comparación con otros estados de la realidad geográfica. Este proceso de abstracción que permite la obtención de conclusiones generalizadoras necesita un filtro continuo de contraste con la realidad que se intenta representar, y es precisamente esta tarea la que es facilitada enormemente, más aún, posibilitada realmente, por la utilización de técnicas de tratamiento digital de imágenes.

Por otro lado, la precisión de los diagramas y mapas obtenidos automáticamente utilizando programas de ordenador no solamente está en relación con la calidad estética de dichas representaciones -hecho nada despreciable, por supuesto-. También, y me parece muy interesante el subrayarlo, la precisión de esta tecnología propicia la utilización de sus productos como instrumentos de medida, sobre todo en el caso de evaluar las distancias que separan dos o más casos en función de una o más variables, lo que tiene una aplicación nada despreciable en los trabajos de clasificación y de definición de tipologías.

#### I.3.4. Algunos ejemplos.

Volviendo un momento al problema que ha motivado estos últimos párrafos, de racionalizar la representación gráfica y cartográfica en Geografía, que es el auténtico móvil de esta tesis y que enlaza con una preocupación más general de formalización del pensamiento y métodos de esta disciplina, tenemos que afirmar que en este trabajo se pretende llegar a la definición de la serie de invariantes que están presentes en toda tarea de diseño de mapas y diagramas y a su resolución mediante programas de dibujo automático. De esta manera se ofrecen al investigador una serie de herramientas de gran potencia, que puede utilizar combinándolas según los objetivos específicos de las representaciones que pretende obtener.

Como mero ejemplo de lo que acabamos de afirmar vamos a comentar tres supuestos.

Uno de los objetivos comunes a cualquier trabajo de representación cartográfica es el dibujo de contornos geográficamente significativos (contornos de países, de regiones, de unidades administrativas, etc...). Pues bien, si aproximamos a cualquier contorno a representar una poligonal cerrada definida por una secuencia de puntos que a su vez se explicitan por sus coordenadas, la tarea de dibujar el contorno es siempre idéntica, cambiando tan solo los datos del problema. El ordenador, en todos los casos, leerá las coordenadas de los puntos que definen la poligonal a representar y gobernará posteriormente un periférico gráfico\* (impresora\*, plotter\*, pantalla de rayos catódicos\*, ...) donde causará la reproducción de la forma deseada, utilizando un conjunto de instrucciones básicas que siempre serán las mismas.

El tramado de áreas es otra tarea común a muchos tipos de representaciones distintas -piénsese por ejemplo en un mapa de coropletas, en un diagrama de barras tramado, en un diagrama de sectores circulares, etc.-. También en este caso es posible y así se ha diseñado, elaborar un único conjunto de instrucciones que, informando al ordenador del polígono que se quiere tramar -definido como hemos apuntado en el párrafo anterior- y el tipo de trama que se elige, de las que ofrece como posibilidad el programa, realice cualquiera de las tareas que hemos señalado más arriba, y cualquiera otra que requiera un tipo de trama para explicitar una variación cualitativa o cuantitativa.

Otra de las posibilidades que se contemplan en esta tesis es la de los gráficos que representan la variación espacial continua de una característica, levantando un bloque diagrama en perspectiva de la superficie tridimensional que la describe. Es éste, también, un conjunto de tareas idénticas, que se pueden formalizar y traducir a un lenguaje matemático, en primera instancia, e informático posteriormente, para que el ordenador pueda elaborar automáticamente, con precisión y rapidez, todos los cartogramas deseados, independientemente de cuáles sean la referencia espacial y la

característica a analizar (altitud, medida pluviométrica, medida termométrica, precio del suelo, densidad de población, nivel de renta, etc...)

#### I.4. PROPUESTA DE UN SISTEMA INTEGRADO DE TRATAMIENTO DIGITAL DE IMAGENES EN INVESTIGACION GEOGRAFICA.

##### I.4.1. La organización de la base de datos.

Tres tareas fundamentales, que se complementan mutuamente, se encuentran involucradas en el desarrollo de un sistema de tratamiento digital de imágenes en investigación Geográfica.

La primera de ellas es la definición de un diccionario de códigos numéricos que se correspondan adecuadamente con la naturaleza de los datos que describen.

La segunda consiste en la representación de los datos físicos por su configuración espacial. La tercera, es la plasmación de las diversas características temáticas que afectan a dichos datos.

Aunque el contenido propio de esta tesis se centra fundamentalmente en el desarrollo de procedimientos que hacen referencia a la tercera de las tareas señaladas, voy a detenerme en la consideración del conjunto, ya que nunca se comprende la importancia real de cada una de las partes de un todo, si no se contempla el objetivo global.

La tarea de codificación es previa a todas las demás y sumamente importante, ya que ella sola condiciona la calidad del sistema. Cuanto más amplio y lógicamente estructurado sea el diccionario de códigos que se define para calificar cada tipo de dato espacial, más interesantes y completas serán las posibilidades de búsqueda y selección de determinados datos. Por ejemplo, ya se entiende que no es lo mismo disponer de un diccionario de códigos para identificación de datos físicos que se correspondan con el sistema de comunicaciones en el que sólo se contemplan las grandes modalidades del transporte (ferrocarril, carretera, líneas de transporte fluvial, transporte marítimo, transporte aéreo, transporte de energía, etc.), que contar con un diccionario de códigos en el que junto a un código genérico que define una determinada modalidad de transporte, se dispone de una serie de subcódigos que se corresponden con una información más detallada (calidad de la vía de transporte, estado de la misma, etc.).

La representación de los datos por su configuración espacial, requiere la grabación de las coordenadas de los puntos que la definen. En el caso de tratarse de un dato puntual, bastará conocer un par de coordenadas. Si pensamos, en cambio, en un dato

lineal o zonal se requerirá la grabación de todas las coordenadas de los puntos que aproximan la poligonal abierta o cerrada, respectivamente, que describe la configuración espacial del dato. Para referirse a este tipo de trabajo se utiliza el término genérico de digitización - también es frecuente utilizar la palabra digitalización-. Existen máquinas electrónicas (digitizadores \*y scanners) que realizan esta tarea de una forma más o menos automática, liberando al hombre de un trabajo que es costosísimo -irrealizable cuando se trabaja en proyectos de digitización de una cierta verdad-, si se efectúa manualmente.

Una vez realizada la descripción digital de la configuración espacial de un determinado dato, se averigua a qué cuadrículas de un sistema general de referencias afecta. Esta conversión facilitará enormemente la recuperación de dicho dato.

La digitización de datos espaciales tiene planteados problemas muy interesantes sin resolver, que están relacionados con los problemas de generalización cartográfica\* y con la necesidad de describir adecuadamente las redes dinámicas (sistemas fluviales, redes de comunicación, etc), como se ha podido comprobar en un congreso celebrado recientemente en Oxford, organizado por la Asociación Cartográfica Internacional (cfr. Cebrián de Miguel /36/).

Finalmente, la plasmación de las características que afectan a los datos espaciales, requiere, por una parte, una labor de documentación exhaustiva que consiste en ir sondeando, y grabando en soportes informáticos, los valores de las características más interesantes, y por otro, el diseño ininterrumpido de programas automáticos de representación gráfica.

A la vista de todo lo anterior, los diversos procedimientos de representación de la variación de características espaciales que se presentan en esta tesis, aunque tienen valor en sí mismos y son utilizables -ya se han realizado diversas aplicaciones con anterioridad a la presentación de esta tesis- en trabajos de Geografía de ámbito restringido, encuentran su pleno sentido cuando se incorporan, como canal de salida de la información, a un sistema integrado de bases de datos espaciales.

Presentamos a continuación, (Fig. I.1), mediante un diagrama sencillo, el flujo pregunta-respuesta coherente con lo que acabamos de señalar, en un sistema integrado de información geográfica.

#### I.4.2. Calidad de la representación y periféricos gráficos utilizables.

Centrándonos ya en las salidas gráficas que debe ofrecer un sistema de tratamiento digital de imágenes en investigación geográfica, nos planteamos qué herramientas básicas de diseño debe poseer para ofrecer al investigador una representación visual -la

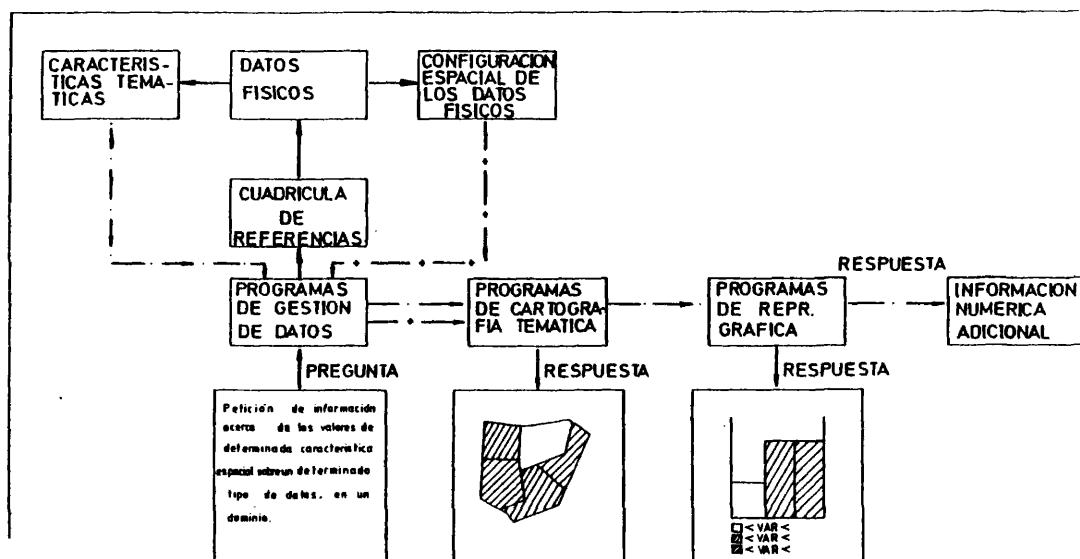


Figura I.I Flujo pregunta-respuesta en un sistema de información geográfica

más sintética- de la realidad que intenta desentrañar y cuáles son los periféricos gráficos más adecuados a utilizar en función de las necesidades y de la relación coste eficiencia, en cada caso.

Por lo que respecta a esta última cuestión, lo más adecuado es disponer las cosas en orden a un aprovechamiento racional de los periféricos gráficos conectables a un sistema de tratamiento gráfico de la información.

Desde esta perspectiva el sistema debe responder, en primera instancia, ofreciendo unos croquis aproximados, realizados con la impresora de líneas, que es el sistema más rápido y económico.

Más adelante, si a la vista de los croquis mediante impresora se exige una representación gráfica de mayor calidad estética y de más adecuada precisión, se pueden confeccionar diagramas o cartogramas utilizando una pantalla de rayos catódicos. Este procedimiento permite, mediante programas interactivos la selección de la representación más adecuada a las necesidades propias.

Solamente cuando se haya llegado a la confección de la representación óptima, se debe ofertar una salida gráfica no perecedera, mediante cualquiera de los sistemas disponibles: plotter térmico, trazador incremental\*, dicomed\*, etc...

Atendiendo ahora a la primera de las cuestiones esbozadas más arriba, tenemos que distinguir dos grandes grupos de tratamientos gráficos: diagramas y mapas.

Los primeros expresan gráficamente las variaciones intrínsecas de una, dos o más características, prescindiendo de la componente espacial -diagramas uní, bi y multivariados-.

Los segundos tienen en cuenta para su representación la localización o distribución de un determinado fenómeno, ya sea físico o georeferenciable, bien en el espacio real, bien en algún espacio relativo o transformado.

En el caso de los diagramas, dado que el nivel de precisión requerido no es muy exigente, lo más adecuado es ofrecer resultados mediante impresora de líneas, ya que en la mayoría de los casos la confección de un diagrama no tiene más que el sentido de lograr una visión sintética e intuitiva de la distribución intrínseca de un fenómeno.

Si se pretende, en cambio, la confección de un mapa, lo más apropiado es ofrecer, paulatinamente, salidas gráficas de creciente calidad y precisión.

#### I.4.3. Definición de tareas.

La petición de una representación gráfica espacial puede tener tres objetivos fundamentales:

- . Localización de un hecho físico o georeferenciable.
- . Explicitación de la variación de una determinada característica física o georeferenciable.
- . Plasmación de flujos de intercambio.

La representación de la localización de hechos físicos o georeferenciables se lleva a cabo mediante la utilización combinada de rótulos y signos convencionales, asociados a las coordenadas geográficas de los puntos en que tiene lugar el fenómeno. En este sentido es vital dotar al sistema de un repertorio de signos con-

vencionales lo más flexible y completo posible.

No se presenta en la tesis ningún procedimiento de este tipo, porque, aunque existen realizaciones aisladas, todavía no está desarrollado definitivamente el paquete de programas de cartografía temática, mediante signos convencionales ponderados, en el que estamos trabajando un equipo de personas.

Por el momento, la minuta de signos ponderados, reproduce automáticamente, comprende los siguientes elementos: círculo, cuadrado, triángulo equilátero, mota circular, mota cuadrada, mota triangular y diagrama circular tramado.

El símbolo aparece centrado sobre el punto en el que se localiza el fenómeno a representar -cuyas coordenadas ha de introducir el usuario previamente-. El tamaño del símbolo lo calcula un subprograma\*, en función de la intensidad del fenómeno en cada punto de la representación.

El tipo de símbolo es algo que debe decidir también el usuario y comunicarlo al programa de dibujo, mediante el código numérico que le corresponda.

Actualmente se trabaja en tres líneas fundamentales:

- . redacción de subprogramas que realizan nuevos tipos de signos convencionales.
- . resolución de los problemas de escala gráfica, rótulos y leyendas, del mapa temático resultante.
- . definición de un programa de control y entrada de datos que facilite la utilización de este tipo de procedimientos a personas sin excesivos conocimientos informáticos.

El resultado final será un sistema flexible, en la línea de los trabajos de Kern y Rushton /89/ y de Monmonier /117/, que permitirá obtener interactivamente una gran variedad de mapas temáticos de gran precisión y calidad.

Por otra parte, la representación de la distribución espacial -variación de una característica, física o georeferenciable-, se puede llevar a cabo mediante dos vías: discreta y continua.

En este sentido es necesario, en primer lugar, dotar al sistema de procedimientos que plasmen la variación discreta de una característica: compartimentación del espacio en polígonos geográficamente significativos y utilización de tramas (representación bidimensional), o levantamiento de prismas de altura proporcional al peso de la variable sobre dichos polígonos (representación tridimensional en perspectiva).

Para hacer frente a la representación de la variación continua de características se requiere disponer de programas de dibujo automático de isolíneas (representación bidimensional), o de bloques diagrama (representación tridimensional en perspectiva), asociados a los programas de interpolación más adecuados a cada procedimiento.

La adaptación de una subrutina\* de dibujo de isolíneas para la representación de la variación continua de una característica espacial, desarrollada por el Virginia Polytechnic Institute and State University /177/, no ha podido concluirse por el momento, aunque será una realidad en muy breve plazo.

Sí se ha resuelto, en cambio, el problema de la plasmación de la variación continua de características espaciales, mediante bloques diagrama en perspectiva isométrica, para cuya elaboración se ha recurrido a los dos métodos de interpolación más usuales en este tipo de procedimientos: ajuste de superficies de tendencia y aproximación numérica. Más adelante, en el capítulo correspondiente a estos procedimientos, se comentan las particularidades de estos dos métodos de interpolación y sus ventajas propias, en función de los objetivos de la representación, utilizando como referencia los trabajos de Mac Dougall /97/ y Tobler /169/.

Finalmente, la plasmación de flujos espaciales de intercambio es una tarea que también ha sido acometida, pero que todavía no se encuentra suficientemente desarrollada. Por el momento, se dispone de dos tipos de procedimientos. El primero, para representar flujos espaciales mediante flechas ponderadas (5). El segundo, para representar áreas de dependencia respecto a un núcleo organizador, mediante radios dirigidos desde cada polo a sus tributarios. (6).

En este sentido, se prevee que, previa una discusión detallada para definir qué tipo de tratamientos gráficos son los más adecuados para plasmar los intercambios dinámicos que se producen entre las diversas compartimentaciones del espacio, se podrá disponer de un sistema de representación de flujos adaptado a las necesidades de la investigación Geográfica -partiendo, por supuesto, de una información espacial almacenada digitalmente-.

#### I.5. DESCRIPCION DEL CONTENIDO DE LA TESIS.

A lo largo de la tesis, con las restricciones que se acaban de señalar, en sucesivos capítulos, se va haciendo frente a los problemas planteados.

Una restricción más hay que tener en cuenta. A pesar de que se ha hablado de múltiples periféricos conectables a un ordenador para la obtención de salidas gráficas: impresora de líneas, panta



lla de rayos catódicos, plotter térmico, dicomed, trazador incremental, etc., en el presente trabajo sólo se aborda la solución de problemas de representación mediante uno de ellos: el trazador.

Dos razones fundamentales justifican esta última restricción.

Por un lado, la definición y desarrollo de programas gráficos que utilizan la impresora de líneas resulta prácticamente innecesaria, en tanto en cuanto estos objetivos ya se encuentran plenamente satisfechos por programas ya implementados en varios Centros de Cálculo universitarios, accesibles a cualquier investigador que tenga interés en su utilización.

Me estoy refiriendo, en primer lugar, a la biblioteca de programas\* estadísticos BMDP, que posee un repertorio de salidas gráficas -curvas y diagramas- muy amplio, que describen la variación intrínseca de una, dos o más características, originales o transformadas, y, en segundo término, al programa de cartografía automática SYMAP. Este último programa, desarrollado por el "Laboratory for Computer Graphics & Spatial Analysis" del "Harvard Graduate School of Design", del que ya se conocen cinco versiones, genera automáticamente cartogramas temáticos que expresan la variación espacial discreta o continua de una determinada característica. Adquirido por primera vez en España por iniciativa de los responsables de la Sección de Cartografía Automática del IGN y del Instituto de Geografía Aplicada del C.S.I.C., hoy es un programa muy difundido entre algunos sectores de la investigación geográfica española y accesible en varios Centros de Cálculo universitarios. Para facilitar, todavía más, la difusión de este programa en los medios universitarios españoles, no solamente a nivel de posgraduados, se ha redactado recientemente, por iniciativa del Departamento de Geografía Humana de la Facultad de Geografía e Historia de la Universidad Complutense, una versión adaptada del manual de uso del SYMAP (Sáez Cala, Vidal Domínguez y Rojo Pérez /148/), que estará disponible en breve para su consulta por alumnos de últimos cursos de licenciatura.

Lo mismo puede decirse respecto de la biblioteca de programas BMDP, cuya descripción -haciendo, por supuesto, mayor hincapié en los programas de tratamiento estadístico- e instrucciones de uso, en versión adaptada a las necesidades de la investigación geográfica, está a punto de publicarse (Santos Preciado y Muguruza Cárdenas /149/).

Por otra parte, la confección de programas gráficos que utilicen otros periféricos distintos del trazador incremental -plotter térmico, pantalla de rayos catódicos, dicomed, etc.-, me ha parecido que es una tarea a acometer cuando este tipo de máquinas esté bastante más difundido que en la actualidad. En los centros universitarios accesibles a la investigación geográfica española resulta difícil encontrar periféricos gráficos de calidad y precisión que no sean trazadores incrementales. Sólo en puntos muy significados y de difícil acceso directo todavía -IGN, Centro UAM-IBM, etc.- se utilizan máquinas más sofisticadas en el curso de

trabajos muy específicos: proyectos de cartografía automática, de finición de bases de datos espaciales, tratamiento de imágenes procedentes de satélites artificiales etc.

Los capítulos en que se articula la tesis responden a una creciente complicación de las tareas a acometer para atender a las necesidades de representación gráfica de la investigación geográfica.

En el capítulo primero: "Elaboración automática de diagramas", se presentan dos programas que realizan dos diagramas de amplio uso en investigación geográfica: la pirámide de población y el diagrama triangular.

Unas páginas más arriba acabamos de comentar que resulta mucho más razonable la utilización de la impresora de líneas para la reproducción de diagramas, frente a cualquier otro periférico gráfico. A pesar de ello, la inclusión de estos dos programas de dibujo mediante plotter posibilita que los gráficos resultantes sean directamente publicables, sin requerirse un redibujo de los mismos -cosa que sería necesaria en el caso de haberse reproducido estos gráficos en una impresora de líneas-.

En otro orden de cosas, este primer capítulo se justifica por cuanto reúne los primeros resultados de cierto interés de mi trabajo en el campo del diseño mediante ordenador, y por el hecho de que la relativa ausencia de complejidad de los programas que lo constituyen facilita una explicación detallada de los fundamentos básicos de las nuevas técnicas de representación gráfica.

El título del segundo capítulo "Cartografía temática en dos dimensiones: MAPAS de COROPLETAS", se refiere a aquello que constituye su parte fundamental, sin duda, pero no la única. También se estudian en este capítulo otros temas interesantes, previos a la construcción automática de un mapa de coropletas.

Así, el capítulo comienza con la consideración de los problemas de definición de un área en términos digitales y de diversas alternativas de solución. A continuación, se plantea y resuelve el rayado de polígonos -herramienta básica en el desarrollo posterior de toda la tesis-. Más adelante, se abordan los problemas de clasificación que están relacionados con la construcción de cualquier mapa de coropletas y se ofrece, como resultado, un conjunto de procedimientos muy flexible.

En el capítulo tercero "Cartografía temática en tres dimensiones: bloques diagrama en perspectiva", se aborda el problema de la obtención de proyecciones isométricas de volúmenes sobre el plano de dibujo.

Todos los programas incluidos en esta parte del trabajo, presentan la variación espacial de una característica física, o la de una característica económicosocial indistintamente. Las dos di

mensiones horizontales se corresponden en este tipo de gráficos con las coordenadas que se refieren a los puntos que definen la zona a estudiar y a los puntos interiores a ella donde se ha medido el valor de la característica, siendo la dimensión vertical de la índole que se estime oportuna (altitud, medida pluviométrica, termométrica o barométrica, valor del precio del suelo, nivel de actividad laboral, etc.).

Los programas presentados en este capítulo se pueden subdividir en dos grandes categorías: los que plasman la variación espacial continua de una característica, y los que expresan gráficamente la variación discreta de un fenómeno sobre un mosaico de zonas individualizadas de estudio. Más concretamente, se presentan en este apartado tres programas, dos de los cuales efectúan la primera de las tareas descritas, realizando el tercero la segunda.

Finalmente, en el capítulo cuarto "La cuarta dimensión: el tiempo. Animación gráfica de bloques diagrama" se presenta un procedimiento de interpolación temporal que, combinado con subprogramas ya definidos previamente en la tesis, permite obtener, mediante plotter, las imágenes necesarias para construir posteriormente, con un filmador, una animación gráfica de la evolución temporal de volúmenes.

A pesar de haber llegado a realizaciones concretas, este campo de la representación, tan interesante en el marco de trabajos que apunten a una consideración simultánea de las variaciones espacio-temporales de los fenómenos, no ha sido agotado, ni mucho menos, en su totalidad. La razón fundamental de ello la ha constituido el hecho de no tener acceso a las máquinas más apropiadas para este tipo de trabajos. A pesar de ello, en el momento presente estoy en contacto con las instituciones apropiadas y con investigadores del campo de la electrónica, para hacer viable un sistema de animación gráfica en tiempo real, asumiendo los últimos desarrollos de la cartografía dinámica (Moellering /115/).

En las conclusiones de este trabajo se exponen, junto con una consideración ponderada de todos los trabajos desarrollados en el marco de esta tesis, las líneas fundamentales de investigación futura que están irremisiblemente relacionadas con el desarrollo de los gráficos por computadora y su aplicación a la investigación geográfica.

#### I.6. ALGUNAS CONSIDERACIONES.

Antes de terminar estas páginas, no puedo dejar de comentar un último aspecto del trabajo que ha configurado esta tesis. Me refiero al hecho de que ha sido desarrollado todo él, si bien contando con libros de referencia y consulta extranjeros en su mayor parte, como puede comprobarse en la bibliografía reseñada, con me

dios existentes en nuestra Universidad y en organismos oficiales de investigación situados en Madrid: Centro de Cálculo de la Universidad Complutense, Centro de Estudios y Sección de Cartografía Automática del Instituto Geográfico Nacional y, con menos incidencia sobre el contenido estricto de la tesis, Cátedra de Matemáticas de la Escuela T.S. de Ingenieros Industriales de la Universidad Politécnica de Madrid.

En todos estos lugares no sólo he podido acceder fácilmente a todas las prestaciones de las máquinas allí emplazadas, sino que he contado en todo momento con el asesoramiento técnico necesario para ir solucionando los problemas de todo tipo que ha habido que superar en el desarrollo del trabajo. Esto último es especialmente notorio en el caso de CCUC, en donde he trabajado en estrecha colaboración con Miguel García Ferrández. Como ya he señalado claramente unas páginas más arriba, sin su ayuda esta tesis no sólo no habría podido concluirse, sino ni siquiera haber sido formulada como proyecto.

Me estoy refiriendo con un cierto detalle a estos aspectos con el fin de demostrar que no es necesario, por lo menos hasta un determinado nivel, acudir a alguna universidad extranjera de prestigio para realizar un trabajo que intente situarse en la línea de uno de los temas que realmente preocupan a nivel mundial en nuestra disciplina en el momento presente.

Con una información actualizada y un diálogo con personas que desde el campo de las ciencias experimentales o de las matemáticas tienen intereses comunes con nuestra ciencia, se pueden ir desarrollando nuevas metodologías y técnicas de tratamiento de datos de carácter autóctono, en vez de tener que importarlas indiscriminadamente del extranjero, con la dependencia económica e intelectual que ello lleva consigo. En este sentido me parece interesante subrayar la actitud de la Geografía francesa, en el campo de la representación gráfica automatizada -que es el que ahora nos ocupa-, en la que junto con la adquisición de paquetes de programas extranjeros, americanos fundamentalmente, se está verificando un importante desarrollo de nuevos sistemas de representación de carácter propio.

Sin necesidad de salir de España, se puede, en el momento presente, acometer la formación de auténticos especialistas en las ramas más tecnificadas de la Geografía, que, reconstruyendo paso a paso toda la historia reciente de la revolución metodológica que se ha verificado en nuestra disciplina, adquieran una cualificación que les permita un diálogo no excesivamente desventajoso con sus colegas de otras nacionalidades.

Una vez alcanzado ese nivel, se puede plantear con garantía de éxito la estancia de estas personas en alguna institución extranjera de calidad, para asimilar -entonces estarán capacitados para hacerlo con profundidad- los últimos avances que se hayan producido en su campo de especialidad y de los que no haya todavía

constancia escrita y detallada en las publicaciones científicas más difundidas.

De todas formas, el desarrollo de proyectos como el que acaba de esbozar no está exento de dificultades y requiere, entre otros extremos, una mayor sensibilización por parte de las autoridades académicas con vistas a facilitar este tipo de trabajos y a lograr una mayor flexibilidad en los planes de estudio que permita con templar la creación paulatina de nuevas materias en los cursos de licenciatura y doctorado, conforme se vayan verificando nuevos avances metodológicos y conceptuales en las respectivas disciplinas.

NOTAS A LA INTRODUCCION

- (1) Ver BENNETT, R.J. y WRIGLEY, N. (eds.), (1980), Quantitative Geography in Britain: Retrospect and Prospect, Londres, Routledg & Kegan Paul, introducción; ESTEBANEZ, J., (1982), Tendencias y problemática actual de la Geografía, Madrid, Ed. Cincel, 144 pp., cap. 7; CAPEL, H., (1981), Filosofía y ciencia en la Geografía contemporánea, Barcelona, Ed. Barcanova, 509 pp., cap. XIII; JOHNSTON, R.J., (1979), Geography and Geographers, Londres, Ed. Arnold, 232 pp., caps. 3 y 4.
- (2) Por la procedencia de las fuentes bibliográficas que se han manejado -americanas e inglesas fundamentalmente-, las realizaciones en este campo llevadas a cabo por geógrafos de otras nacionalidades han sido, sin duda, infravaloradas -a excepción del caso francés, ya que la relación directa con el Laboratorio de Cartografía Temática de la Universidad Louis Pasteur de Estrasburgo ha permitido disponer de una información actualizada y completa-.
- (3) Geography Program Exchange, Computer Institute for Social Science Research, 515 Computer Center, Michigan State University, East Lansing, Michigan, 48823, USA.
- (4) Una prueba de ello es la escasa participación de geógrafos españoles -tan sólo tres- en la II Conferencia Internacional de Bases de Datos en Humanidades y Ciencias Sociales, celebrada en Madrid en junio de 1980. Las referencias bibliográficas, por otra parte, también son muy reducidas (Barredo Risco /6/, Bosque Sendra y Cebrián de Miguel /21/ y Herrero García, Bosque Sendra y Cebrián de Miguel /76/).
- (5) Este trabajo ha sido realizado por Beatriz Jiménez Blasco.
- (6) Este trabajo ha sido realizado por Miguel García Ferrández y Juan Antonio Cebrián de Miguel.

CAPITULO 1. ELABORACION AUTOMATICA DE DIAGRAMAS.

	Págs.
1.1. INTRODUCCION .....	29
1.2. TRATAMIENTO DIGITAL DE IMAGENES EN TECNOLOGIA VECTORIAL .....	29
1.2.1. Descripción raster y descripción vectorial de una imagen.....	29
1.2.2. Estructura básica de los programas de representación gráfica vectorial .....	33
1.2.3. Justificación de estas técnicas en trabajos de diseño gráfico y cartográfico .....	34
1.3. DIBUJO AUTOMATICO DE PIRAMIDES DE POBLACION SIMPLES, COMPUESTAS O SUPERPUESTAS. EL PROGRAMA PIRAMID.....	36
1.3.1. Definición en términos precisos del problema de la representación .....	37
1.3.2. El problema de la agregación de los datos....	39
1.3.3. La transformación de los datos originales....	40
1.3.4. La solución standard al dibujo de pirámides..	40
1.3.5. La realización automática de pirámides compuestas .....	51
1.3.6. La superposición de dos pirámides de población .....	59
1.3.7. Información adicional, en función de las necesidades del usuario .....	63
1.3.8. Flexibilidad de los parámetros fundamentales de dibujo .....	65
1.3.9. Requisitos y organización de los datos para uso del programa. ....	66
1.3.10. Listado del programa. ....	66
1.3.11. Algunos ejemplos .....	75
1.4. DIBUJO AUTOMATICO DE DIAGRAMAS TRIANGULARES. EL PROGRAMA TRIAN.....	75
1.4.1. Los términos de la representación .....	75
1.4.2. Limitaciones actuales del procedimiento .....	80
1.4.3. Construcción automática del diagrama triangular	81
1.4.4. Control de la legibilidad del gráfico resultante	82
1.4.5. Listado del programa .....	83
1.4.6. Algunos ejemplos .....	87

### 1.1. INTRODUCCION.

La tarea de elaboración de programas que realizan automáticamente diagramas de uso frecuente de investigación geográfica ya ha sido avanzada en el curso en una Memoria de Licenciatura presentada en esta Facultad. (Martínez de Sola /105/). En esa misma línea de trabajos (1) se presentan ahora dos programas que realizan diagramas muy utilizados: la pirámide de población y el diagrama triangular (2).

El primero de ellos hace ya tiempo que ha sido redactado y está siendo objeto de un amplio uso desde entonces. De hecho, se hace referencia a él en la Memoria a que acabo de aludir como un programa accesible en el Centro de Cálculo de la U.C.M., que libera de una tarea muy frecuente, que resulta considerablemente engorrosa cuando se pretende describir con este procedimiento gráfico la estructura por edad y sexo de un número considerable de grupos de población.

Es a este tipo de programas a los que se refiere en parte el título de la tesis: Diseño mediante ordenador EN RESOLUCION DE PROBLEMAS DE REPRESENTACION GRAFICA y cartográfica en Geografía.

Se presentan en el primer capítulo por dos motivos. En primer lugar porque han constituido mi iniciación en la resolución de problemas en este campo de la investigación. En segundo término por que por su relativa simplicidad pueden ser más fácilmente comprendidos, como ejemplos de la aplicación del tratamiento digital de imágenes a tareas instrumentales en Geografía.

### 1.2. TRATAMIENTO DIGITAL DE IMAGENES EN TECNOLOGIA VECTORIAL.

#### 1.2.1. Descripción vectorial y descripción raster de una imagen.

En el momento presente se dispone de dos vías de aproximación a la descripción numérica de una determinada imagen: el modo vectorial y el modo raster. En el primero de los casos, sólo se consideran las partes significativas de la imagen: puntos, contornos lineales o superficiales que la configuran. En el segundo modo se considera descompuesta la totalidad de la superficie de la imagen en una serie de partículas elementales (en general cuadrados), considerando para cada una de ellas un código relacionado con la clase de dato que existe en ella o con la inexistencia de información en la misma. De la utilización de uno u otro modo se derivan numerosas consecuencias por lo que respecta a la naturaleza de los procesos de tratamiento necesario (cfr. Herrero García y Cebrián de Mi-



guel /77/).

En el caso de la presente tesis, la razón fundamental que ha causado el que a lo largo de toda ella no se utilice otro que el modo vectorial ha sido el hecho de disponer de un periférico gráfico -plotter incremental- que presupone este tipo de descripción numérica. Si, por el contrario, se hubiera trabajado con otro tipo de periféricos -impresora de líneas, transformador cinta/película o impresora electrostática, por ejemplo- el modo de descripción elegido hubiera sido el raster.

Centrándonos ya, por tanto, en el modo vectorial de descripción numérica de imágenes, es necesario establecer una serie de consideraciones previas acerca de la descomposición de una imagen en sus elementos constitutivos, para su tratamiento mediante ordenador.

Toda figura, por complicada que sea, puede ser considerada como un conjunto de líneas o, hablando más propiamente, segmentos de línea.

A su vez, toda línea es un conjunto de puntos unidos por tramos rectos (cuanto mayor sea el número de puntos que consideremos, la poligonal que los relacione será una aproximación más acertada en la línea a representar).

Finalmente, si logramos establecer un orden dentro de todos los puntos del plano, cada punto podrá ser conocido por el número que le corresponde, en el orden.

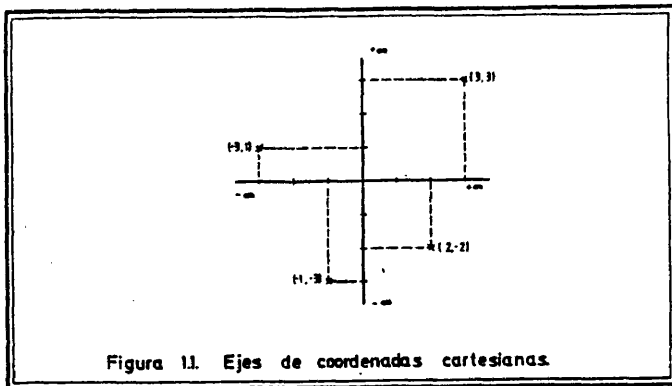
En el momento en que se puede establecer un orden dentro de todos los puntos del plano, el ordenador, que es capaz de almacenar electrónicamente cantidades numéricas, podrá conocer en abstracto cualquier figura como el conjunto de valores numéricos que se corresponden con los puntos del plano que constituyen dicha figura.

El instrumento para poder reconocer numéricamente los puntos del plano es  $R^2$ , o sea el conjunto de pares de números reales. Gracias a esta abstracción matemática, que se debe fundamentalmente a Fermat y Descartes, es posible la descripción numérica de figuras usual en Geometría Analítica.

Si sobre un plano dibujamos un par de ejes, que se cruzan en ángulo recto con una escala real en cada uno de ellos, a cada punto del plano le corresponde un par y sólo uno de valores, que son la magnitud de sus proyecciones ortogonales sobre cada una de las dos escalas reales definidas (Fig. 1.1.),

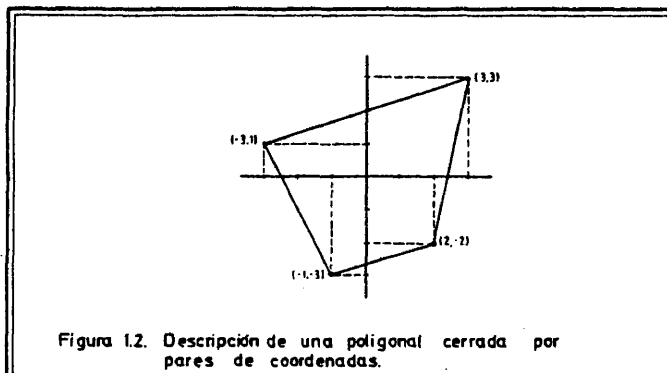
Los dos números reales así obtenidos se llaman coordenadas cartesianas del punto.

Continuando con el ejemplo que hemos definido en la figura precedente, el ordenador puede conocer la poligonal cerrada de la figura 1.2., como el conjunto de pares de números reales siguientes-



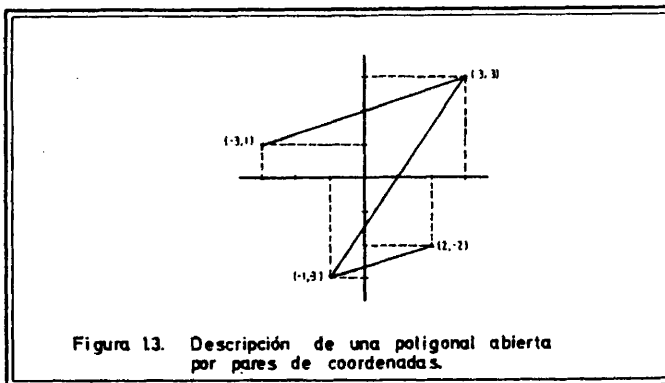
tes, ordenados de esta forma:

$(-1,-3)$ ,  $(2,-2)$ ,  $(3,3)$ ,  $(-3,1)$ ,  $(-1,-3)$ .



De igual manera, la poligonal abierta de la figura 1.3, puede ser reconocida en abstracto por el ordenador como la siguiente sucesión de pares de números reales:

$(-3,1)$ ,  $(3,3)$ ,  $(-1,-3)$ ,  $(2,-2)$ .



Para procesar una imagen mediante ordenador es necesario describirla como conjunto de relaciones existentes entre puntos del plano.

En función de las relaciones que se pueden establecer entre los puntos del plano, se puede definir una jerarquía de unidades elementales de representación gráfica.

**Puntos aislados:** Elementos que para los efectos del gráfico no tienen ninguna relación con ningún otro.

**Líneas:** Conjuntos de puntos relacionados, cuyo grafo de relaciones es una poligonal abierta.

**Superficies:** Conjuntos de puntos del plano en el interior de una poligonal cerrada.

Por otra parte, cada uno de estos tipos de unidades tiene unas características propias que es necesario explicitar.

Cuando sólo se consideran dos características de cada uno de estos tipos de unidades, las dos dimensiones del plano, convenientemente significadas, son suficientes para describirlas adecuadamente. Es el caso de los gráficos simples de distribución de elementos en función de dos características, de las curvas de evolución temporal de una determinada característica en un determinado ámbito y de la delimitación de zonas en las que se produce una determinada combinación de dos características. La única información adicional que requiere este tipo de representación es la rotulación adecuada de los ejes y sus divisiones, para que refleje el signifi

cado y la naturaleza de la variación que se ha asignado a cada una de las dos dimensiones del plano.

En el momento en que se quiere expresar al mismo tiempo sobre un único gráfico la variación conjunta de más de dos características, afectando a varias unidades de las anteriormente definidas es necesario recurrir a algún tipo de artificio para representarla sobre un plano. Estos nuevos instrumentos de diseño son fundamentalmente de dos tipos: rótulos y signos convencionales incluidos en el dibujo, y sistemas de proyección de espacios de más de dos dimensiones sobre un espacio bidimensional, creando la ilusión óptica de un volumen tridimensional visto en perspectiva.

#### 1.2.2. Estructura básica de los programas de representación gráfica vectorial.

A la vista de todo esto se puede comprender que los programas de diseño gráfico mediante ordenador sean conjuntos de instrucciones de lectura de datos y de cálculo de coordenadas en función de éstos con vistas a causar una imagen mediante el recorrido de puntos -definidos por pares de coordenadas- por una pluma que puede moverse entre cada par de ellos levantada (sin pintar) o bajada (pintando). Una imagen que sea una explicitación visual, sintética y más elocuente de los datos numéricos a los que hace referencia.

Existe todo un campo de diseño mediante ordenador que se refiere a tareas que requieren como datos de entrada series estadísticas y que, en función de ellas, elaboran representaciones visuales de sus características intrínsecas. Para la elaboración de estos diagramas, de complejidad muy variable -acorde con el volumen de información que se requiere asociar a un determinado dibujo-, el programa de diseño comienza por asignar las sucesivas dimensiones de un espacio euclídeo a las características que están descritas en la serie estadística, procediendo a definir a continuación una escala gráfica adecuada a los valores que asume la característica en los diversos elementos de la muestra, para optimizar su representación.

Si el número de características a representar es igual o inferior a dos, las dos dimensiones ortogonales del plano son suficientes, en caso contrario será necesario recurrir a diversos procedimientos para obtener una imagen que sintetice una información más compleja (proyecciones de espacios de más de dos dimensiones sobre un espacio bidimensional, utilización de signos convencionales, de rótulos, etc.).

Se consigue de esta forma dar significado a todos los puntos del plano y se dispone de una métrica para reconocer aquellos que representan una determinada combinación de características. A partir de ese momento, el ordenador puede reconocer puntos aislados, conjuntos lineales de puntos, o superficies, que son compartimentaciones significativas del espacio definido.

Los programas de ordenador que tienen por objeto la realización de cartogramas automáticamente requieren, además, como datos

iniciales, junto a la lista de valores de las características a re presentar, que afectan a determinados elementos del plano (puntos, líneas o áreas), la información necesaria para reconocer digital mente dichos elementos -par de coordenadas para los elementos pun tuales; series de pares de coordenadas que definen una poligonal abierta para elementos lineales; series de pares de coordenadas que definen una poligonal cerrada para reconocimiento de elementos su perfciales-.

Esta tarea que se conoce con el nombre de digitización, cuan do ha de ser realizada a mano es sumamente costosa -de ahí la nece sidad insoslayable de ir dotando de digitizadores, o de otro tipo de sensores gráficos, a las instituciones de investigación que tra bajan en el campo de las ciencias espaciales- (3).

Quiero hacer hincapié en este hecho, ya que los programas de cartografía automática que presento sólo tendrán una aplicación con siderable en el caso de que se disponga de amplias bases de datos con referenciación espacial, implementadas mediante procesos auto máticos o semiautomáticos -utilizando digitizadores o scanners-. Mien tras tanto, sólo podrán atender a objetivos concretos de investiga ción, en los que, por el interés de los usuarios de estos prográ mas, se realice una conversión digital de la configuración espacial básica -bien a mano, que será lo más frecuente, o mediante algún sensor gráfico- (4).

En trabajos de representación cartográfica las dos dimensio nes del plano son previamente asignadas a las dos componentes espa ciales, lo que nos permite reproducir una imagen geográficamente significativa. Por ello, cualquier otra característica que afecte al terreno, fuera de su configuración básica, ha de representarse mediante algún tipo de artificio que consiga plasmar sobre el espa cio bidimensional del plano la variación de una tercera, cuarta, quinta u otra cualquiera magnitud.

### 1.2.3. Justificación de estas técnicas en trabajos de diseño grafi co y cartográfico.

Según lo que acabamos de ver resulta necesaria una transfor mación previa del material gráfico para su procesado en un ordena dor (descomponer la figura en segmentos de líneas y las líneas en conjuntos de puntos). Si se piensa que esto ha de hacerse manual mente, se comprende que se rechacen estos procedimientos ya que es to supondría un trabajo improbable y costosísimo.

La utilización del ordenador en procesos de diseño, sólo es justificable cuando el trabajo realizado por la máquina compensa realmente el realizado por el hombre para que éste pueda trabajar.

Pues bien, en el momento presente resulta claramente ventajo sa la utilización de ordenadores en todo tipo de diseño por tres mo tivos fundamentales:

1°. Por la automatización del trabajo de análisis de las fi guras.

Existen máquinas, como ya hemos comentado, que, con una intervención mínima, cada vez más reducida en tiempo y esfuerzo, convierten una imagen en su descripción digital, generando "chorros" de pares de números reales (coordenadas).

Estas máquinas (digitizadores), auténticos sensores, constituyen una apertura hacia el exterior de la capacidad de reconocimiento del ordenador. Análogamente a como el cerebro humano se relaciona con el medio circundante mediante las capacidades sensitivas del hombre, un ordenador puede obtener un reconocimiento apropiado a sus procedimientos de discurso de las imágenes externas, mediante la utilización de sensores gráficos.

## 2°. Por la capacidad constructiva de imágenes del ordenador.

La información numérica almacenada en el ordenador puede ser manipulada posteriormente por programas de diseño gráfico con vistas a la reproducción de la imagen original. Pero también existen tareas de diseño auténticamente constructivas.

Muchas de las tareas encomendadas al diseño gráfico mediante ordenador superan la mera reproducción de una imagen previa, diseñada en sus componentes elementales para su comprensión por una máquina digital.

Es más, la auténtica aportación de la incorporación de los ordenadores al diseño gráfico es la capacidad de éstos de construir imágenes originales en función de determinadas propiedades del espacio, que puedan ser explicitadas mediante una relación de tipo numérico.

Por ejemplo, si nos proponemos obtener un mapa de coropletas de los niveles de renta en España, utilizando como base de agregación las provincias españolas, resulta imprescindible obtener una representación abstracta, en forma numérica, de los contornos provinciales. Es esta una tarea de análisis de la imagen original, que es necesario efectuarla, bien manualmente, obteniendo sobre un retículo ortogonal el valor de las coordenadas de un número suficiente de puntos de cuya unión mediante segmentos rectilíneos resulte una aproximación de la realidad a representar, bien automáticamente mediante la utilización de un sensor gráfico.

Este trabajo no puede realizarse de otra manera, ya que no existe posibilidad alguna de que el ordenador fragmente la superficie de la península Ibérica en espacios administrativos -no existe relación alguna que permanezca constante entre las dos coordenadas de todos los puntos que definen la poligonal que aproxima un contorno provincial-, y por tanto debe conocer la forma del encasillado administrativo como un dato de partida.

Ahora bien, a partir de ese momento, si el ordenador conoce las propiedades del espacio en el interior de cada fragmento -lista de valores de renta asociados a cada contorno provincial-, puede

automáticamente reproducir dentro de cada ámbito una trama de densidad proporcional al valor de la renta en cada compartimento. Es ésta sin duda, una tarea creativa, tal como la hemos definido más arriba.

De igual manera, utilizando otro tipo de representación, el ordenador podría haber levantado sobre cada contorno provincial un prisma de altura proporcional al valor de la renta dentro de él, o bien haber efectuado deformaciones topológicas de los contornos, o cualquier otro tipo de imagen que explicita las variaciones espaciales de dicha característica.

### 3°. Por la velocidad de respuesta gráfica del ordenador.

Es un hecho incontestable la auténtica explosión verificada en la capacidad de obtención de salidas gráficas de toda índole, por la utilización de esta tecnología.

La configuración original del espacio queda almacenada en un soporte informático (tarjetas, cintas, discos...) y se puede recurrir a ella para obtener las más diversas imágenes de las distribuciones de múltiples características, recurriendo además a muy diversos sistemas de representación, sobre dicho espacio.

En sistemas de tratamiento gráfico completos, las diversas imágenes obtenidas se reproducen en pantallas de rayos catódicos a gran velocidad, seleccionando y conservando copia no precedera de las auténticamente representativas.

### 1.3. DIBUJO AUTOMÁTICO DE PIRÁMIDES DE POBLACION SIMPLES, COMPUESTAS O SUPERPUESTAS. EL PROGRAMA PIRAMID.

Mis primeros trabajos en el campo de los gráficos por computadora tuvieron por objeto la ilustración de resultados procedentes de tratamientos numéricos para establecimiento de proyecciones de población y utilizaban como periférico gráfico la impresora de líneas (5).

Con estos antecedentes, al entrar en contacto con las enormes posibilidades de representación gráfica de otros periféricos distintos de la impresora de líneas -me estoy refiriendo concretamente al trazador de curvas incremental- me planteé la posibilidad de redactar un programa que realizara pirámides de población de mayor precisión y calidad estética, y que fuera mucho más versátil que el que ya estaba funcionando.

Antes de comenzar la tarea, me planteé detenidamente cuáles son las prestaciones fundamentales que una persona que necesita representar gráficamente la estructura por edad y sexo de una población, puede requerir, en función de sus objetivos propios.

### 1.3.1. Definición en términos precisos del problema de la representación.

En el objetivo general de dibujo de una pirámide de población están involucrados varios problemas parciales, que conviene explicitar debidamente.

1. El primero de ellos, que no está directamente relacionado con la elaboración del dibujo, es el problema de los datos a representar. Se podría haber prescindido de él y limitar por tanto, las posibilidades de uso del programa.

Por ejemplo, dado que la agregación de la población en grupos quinquenales es la más frecuente, se podría haber decidido elaborar un programa que efectuara pirámides de población de grupos quinquenales de edad. Este programa podría ser interesante, pero tendría unas limitaciones de uso muy claras.

Resulta, en cambio, mucho más lógico disponer las cosas de tal manera, que, generalizando el programa, éste sea capaz de representar cualquier pirámide de población, sea cual sea el nivel de agregación de los datos.

Yendo más allá, se puede pretender liberar al usuario del programa del trabajo, facilitado probablemente por el uso de una calculadora, de tener que agregar los datos, partiendo de los grupos de edad anuales -que es la información que aparece normalmente en las publicaciones estadísticas más difundidas- y de calcular el porcentaje--o cualquier otro tipo de medida ponderada- que representa una determinada cohorte respecto del total de la población.

El resultado de todas estas consideraciones previas es la posibilidad de utilizar este programa de dibujo, para representar cualquier estructura por edad y sexo, independientemente de la naturaleza de los datos, con tal de que el usuario la defina mediante descriptores sencillos y manifieste previamente qué tipo de manipulaciones desea que se hagan con los datos, antes de presentarlos.

En esta definición de datos por parte del usuario está incluida la posibilidad de que éstos se refieran a pirámides de población compuestas o a la superposición, para comparación, de varias pirámides sobre unos mismos ejes de referencia.

2. Otro problema a considerar es el de las dimensiones del dibujo a obtener.

Una primera solución del problema la constituyó el definir las dimensiones del dibujo mediante la longitud de la base de la pirámide, cuyo valor debía decidir el usuario, en función de sus propias necesidades. Conocido este dato, el ordenador calculaba automáticamente, según unas instrucciones standard, todos los de-



más valores necesarios para el dibujo: escala horizontal, altura de la pirámide, escala vertical, etc...

La prueba del programa con datos reales de estructura por edad y sexo para presentación de resultados de investigaciones recientes demostró que era necesario -además de la versión standard- ofrecer al usuario la posibilidad de controlar independientemente las dos dimensiones del dibujo y la escala horizontal del mismo, y así se ha hecho.

También se han dispuesto las cosas de manera que, una vez obtenido un dibujo en las proporciones deseadas, se puedan realizar ampliaciones o reducciones del mismo.

3. Dado que en la actualidad no existe un criterio único para elaborar pirámides de población, en el sentido de que algunos autores consideran que es la longitud del rectángulo que representa las dimensiones de un determinado grupo de edad la que debe ser proporcional a los efectivos de dicho grupo, y otros, en cambio, consideran que debe serlo la superficie, el programa está diseñado para dibujar la pirámide atendiendo a cualquiera de los dos criterios, optando el usuario por uno de ellos.

4. El objetivo de representar pirámides compuestas de población está resuelto recurriendo a un conjunto de tramas standard, que permitan la representación de la clasificación de la población de cada cohorte en más de dos grupos.

Se han hecho pruebas con pirámides compuestas, en las que dentro de cada cohorte se consideran tres grupos de población y el resultado es claramente satisfactorio.

5. Para representar varias pirámides superpuestas, se ha tenido en cuenta que difícilmente el resultado constituye un gráfico legible, cuando el número de pirámides es superior a dos. En primera instancia se resolvió el problema pintando cada una de las pirámides en diferente color -esto era posible trabajando con el plotter de dos plumas del IGN-, pero se ha recurrido a otra solución por dos motivos. Una de las dos razones a que me refiero es la consideración de que, con mucha frecuencia, es difícil editar los textos en que se vuelcan resultados de investigación a más de un color. Por otro lado, también ha influido en la necesidad de buscar una solución distinta a este problema el hecho de que el plotter del Centro de Cálculo de la Universidad Complutense -que es el realmente accesible a los profesores y estudiantes de esta Facultad- sólo tiene una pluma.

El resultado final ha sido recurrir a dos tramas automáticas, una de las cuales significa que el mayor desarrollo dentro de una cohorte corresponde a una de las pirámides, siendo la otra signo de la situación opuesta.

6. Finalmente, el usuario del programa puede requerir la impresión de alguna información adicional: rótulos para identi-

ficar el ámbito y la fecha a la que corresponde la pirámide; leyenda de tramas -en el caso de representación de pirámides compuestas o superpuestas-; escala vertical que representa los intervalos en los años de nacimiento de los individuos de las diversas cohortes, etc...

Una vez diferenciadas claramente las tareas parciales a resolver, lo más coherente ha sido diseñar una subrutina para cada problema en cuestión y, finalmente, articularlas mediante un programa principal lo más flexible posible.

### 1.3.2. El problema de la agregación de los datos.

Se confía a la subrutina AGREGA el problema de agrupar los datos referentes a la clasificación de la población por sexo y edad.

Utiliza como información de entrada la lista de valores de los efectivos masculinos y femeninos de cada grupo de edad y dos parámetros que define el usuario: nivel de agregación de los datos que presenta al programa y nivel de agregación con el que desea que sean tratados.

El texto de la subrutina, como puede verse en el anexo final, es muy breve -el trabajo que realiza es muy simple-, no obstante estimo que es realmente útil haber considerado esta posibilidad del programa. Gracias a ella el usuario puede desentenderse de otro trabajo que no sea el de grabar en algún soporte (tarjeta, cinta, disco, etc...), la información que recogen las tablas de las publicaciones sobre estructura por edad y sexo de las poblaciones (nacional, provincial, ciudades más importantes, etc.), y añadir a esta información dos descriptores numéricos: amplitud en años de los grupos de edad de los datos de base y amplitud en años de los grupos de edad con que se desea trabajar.

El usuario, si la tabla estadística de que dispone le proporciona información sobre efectivos masculinos y femeninos en cada grupo de edad anual, pero quiere obtener gráficos de esta distribución considerando grupos quinquenales, no tiene más que grabar los datos de dicha tabla e informar al ordenador de que se trata de grupos de edad anuales (el primer descriptor=1) y de que pretende que los grupos resultantes sean quinquenales (el segundo descriptor=5).

Son factibles todas las combinaciones posibles, con la única limitación, lógica por otra parte, de que la amplitud en años de los grupos resultantes ha de ser un múltiplo de la amplitud en años de los grupos de partida.

Si se desea que el programa respete el nivel de agregación de los datos de entrada, los dos descriptores han de tener el mismo valor: amplitud en años de los grupos de edad en que está clasificada la población.

### 1.3.3. La transformación de los datos originales.

Cada vez es menos frecuente representar los volúmenes absolutos de cada cohorte dentro del conjunto de una determinada población, recurriéndose a algún tipo de medida ponderada que haga factible la comparación de varias distribuciones entre sí.

La ponderación más frecuente es calcular el tanto por ciento, del total de la población, que representa una determinada cohorte. Por esta razón se han diseñado dos breves subrutinas, PORCEN y PORCE2, para transformar todos los datos en porcentajes.

Finalmente, si se desea hacer cualquier otra transformación de los datos, y el usuario tiene capacidad para redactar la versión informática del objetivo que pretende -o conoce alguna a la que recurrir para llevar a cabo esta tarea-, tiene la posibilidad de realizarlo recurriendo a un optativo determinado y confeccionando su propia versión de las subrutinas TRANFR o TRANF2, sabiendo que dispone, además de los dos vectores\* en los que están almacenados los valores absolutos de los efectivos de población clasificada por edad y sexo, de dos vectores de dimensiones semejantes para utilizarlos como espacio de trabajo para cuantas manipulaciones sean necesarias.

Hasta aquí nos hemos limitado únicamente, a considerar y resolver informáticamente todos los problemas, de recogida de datos y de preparación de los mismos, previos a la tarea de dibujo de la pirámide. Se trata de cuestiones triviales, por supuesto, pero no quiero dejar de reseñar la gran utilidad que supone disponer de unas herramientas de esta índole.

### 1.3.4. La solución standard del dibujo de pirámides.

Cuando se comienza a trabajar en el diseño de procedimientos informáticos hay una tendencia casi inconsciente, que sólo se corrige cuando se tiene una cierta experiencia, de reducir al mínimo el número de datos que se exigen que defina el usuario, con la pretensión de demostrar claramente la gran potencia del ordenador en resolución de problemas de cálculo.

Existe la pretensión de asombrar al usuario presentándole unos resultados complejos, a partir de un mínimo de información que se le pide que introduzca en la máquina. Este objetivo es fácil de lograr, por escasos que sean los conocimientos informáticos de la persona que redacte el programa, ya que, normalmente, el usuario desconoce los procedimientos más simples.

Posteriormente, si prevalece la honradez y el afán de poner auténticamente a disposición de un amplio número de usuarios los propios programas, se llega a la conclusión de que es neces

rio ofrecer a la persona que va a utilizarlos una amplia gama de mecanismos de control.

Así las cosas, se comprende fácilmente que todo programa debe tener, junto a una versión standard -en la que se efectúa automáticamente toda una cadena de operaciones complejas a partir de una información de entrada escueta-, un amplio espectro de optativos mediante los cuales el usuario puede controlar aquél, si no le resulta satisfactoria ésta.

Vamos ahora a detenernos en la explicación de cómo funciona la versión standard de la subrutina de dibujo PIRAM y de qué información debe conocer previamente.

#### 1.3.4.1. Cálculo de los parámetros de dibujo.

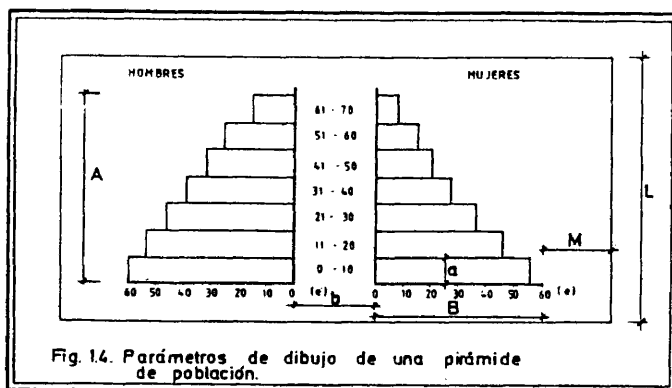
1. Los datos de partida necesarios son los siguientes:
  - Número de grupos de edad que se consideran. (NG).
  - Amplitud en años de los grupos de edad. (NA).
  - Valores absolutos -o medidas ponderadas- de los efectivos de cada cohorte, clasificados por sexos. ( $PM_{ij}$  y  $PF_{ij}$ ,  $i$ =grupo de edad).
  - Dimensión en cm. del lado del cuadrado al que se va a ajustar el dibujo de la pirámide. (L).
  - Valor en cm. del margen de dibujo. (M).

Como puede apreciarse, las únicas opciones que tiene el usuario para definir las características de la pirámide que quiere obtener, es determinar las dimensiones generales del dibujo y los márgenes desde el dibujo al cuadrado en que se enmarca. A partir de ahí el programa calcula automáticamente las restantes características del gráfico que necesitan ser explicitadas.

2. Esquemáticamente, podemos afirmar que los elementos que definen el dibujo de una pirámide son los siguientes (Fig. 1.4.):

- Dimensión de los ejes horizontales (B): que debe ser la misma para ambos sexos y tal que tenga un desarrollo igual o ligeramente superior al escalón más prominente, para poder realmente interpretar la disposición del conjunto.

- Escala de los ejes horizontales (e): que es una relación de proporción entre la unidad de medida de la población en cada grupo de edad y sexo -valor absoluto o medida ponderada- y la unidad de medida longitudinal (mm, cm, dm, o cualquiera otra que se considere), en función de la cual se rotulan los ejes horizontales, situando sobre ellos la disposición y valores de las divisiones.



nes fundamentales.

- Altura de la pirámide. (A).

- Altura de los escalones. (a): que, lógicamente, es la altura total de la pirámide, dividida por el número de grupos de edad que se consideran.

- Escala vertical. (e'): que indica los grupos de edad a los que se refieren los escalones de la pirámide, mediante rótulos que se sitúan en el espacio que separa los dos lados de ésta (b).

En 1. hemos definido los datos del problema, y en 2. las incógnitas que es necesario calcular en función de ellos.

Sumariamente, para no detenernos excesivamente en este punto, podemos describir el procedimiento de resolución.

I. Asumimos que el margen que separa el dibujo de la pirámide del cuadrado que la enmarca es igual a la separación entre los dos lados de la pirámide.

$$b = M$$

II. Lógicamente de esto puede deducirse fácilmente que la dimensión de los ejes horizontales se puede calcular en base a la siguiente ecuación:

$$B = \frac{L - (3.M)}{2}$$

III. A continuación y basándonos en la apreciación de Pressat de que la altura de una pirámide de población debe ser aproximadamente  $2/3$  de la base, podemos calcular A

$$A = 2/3 \cdot ((2 \cdot B) + b)$$

IV. La altura de los escalones es función de la altura de la pirámide y del número de grupos de edad.

$$a = A/NG$$

V. Los rótulos que definen la escala vertical de la pirámide (e'), están compuestos por un límite inferior (INF) y por un límite superior (SUP), y ambos están en función de la amplitud en años de los grupos de edad (NA) y del número de orden de cada grupo, ordenados éstos de menor a mayor edad (I)

$$INF = (I-1) \cdot NA$$

$$SUP = (I \cdot NA) - 1$$

Podemos hacer un par de pruebas para comprobar que esta relación es cierta y constante.

Supongamos en primer lugar que estamos pintando una pirámide de grupos quinquenales de edad y que vamos a situar el rótulo del tercer escalón:

$$INF = (3-1) \cdot 5 = 2 \cdot 5 = 10$$

$$SUP = (3 \cdot 5) - 1 = 15 - 1 = 14$$

Consideramos ahora que vamos a situar el rótulo del quinto escalón de una pirámide de grupos bianuales de edad:

$$INF = (5-1) \cdot 2 = 4 \cdot 2 = 8$$

$$SUP = (5 \cdot 2) - 1 = 10 - 1 = 9$$

En ambos casos vemos que la relación es cierta.

VI. Finalmente el problema de la escala de los ejes horizontales y de su posterior rotulación en función de ella, se resuelve acudiendo a las subrutinas: SCALE y AXIS, del software \* funcional del sistema Calcomp (cfr. Martínez de Sola /105/), que efectúan este objetivo automáticamente, conociendo la longitud de los ejes horizontales (B) y el conjunto de valores a representar ( $PM_i$  y  $PF_i$ ,  $i=1, NG$ ).

La subrutina SCALE, conociendo la longitud del eje horizontal (B) y la lista de valores a representar ( $PM_i$  y  $PF_i$ ,  $i=$  grupo de edad) calcula automáticamente la relación unidad de medida lineal/ unidad de medida de la variable -factor de escala- pertinente, para

que se puedan representar los valores de la lista en el espacio que se les ha asignado previamente.

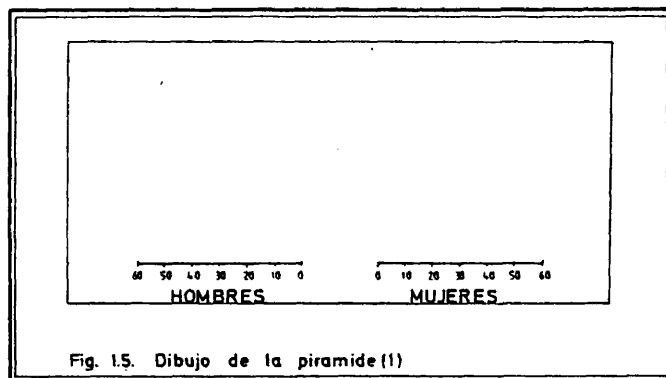
Más concretamente, la subrutina SCALE calcula el número de unidades de medida de la variable que se corresponden con un cm. del plano.

A continuación, la subrutina AXIS, conociendo el factor de escala calculado previamente por SCALE, rotula coherentemente los ejes, situando, centrados respecto de éstos, los literales que expresan el nombre de la variable que se está representando -en el caso concreto de la pirámide de población, los rótulos de "HOMBRES" y "MUJERES" al pie de los ejes izquierdo y derecho, respectivamente-.

#### 1.3.4.2. Dibujo de la pirámide.

Hasta ahora hemos descrito el procedimiento que ha de seguirse para derivar de la información inicial, los valores de los parámetros fundamentales que definen el dibujo de una pirámide concreta. Vamos a continuación a explicitar el procedimiento -que es iterativo- necesario para obtener el gráfico deseado a partir de la información de entrada a esta subrutina y de los parámetros fundamentales del dibujo.

Por el momento, mediante la llamada a las subrutinas SCALE y AXIS, se ha obtenido el siguiente resultado gráfico (Fig. 1.5.).



Ahora vamos a ver la forma de resolver el problema de dibujo de los ejes verticales, su rotulación coherente y de dibujo de los escalones correspondientes a cada grupo de edad.

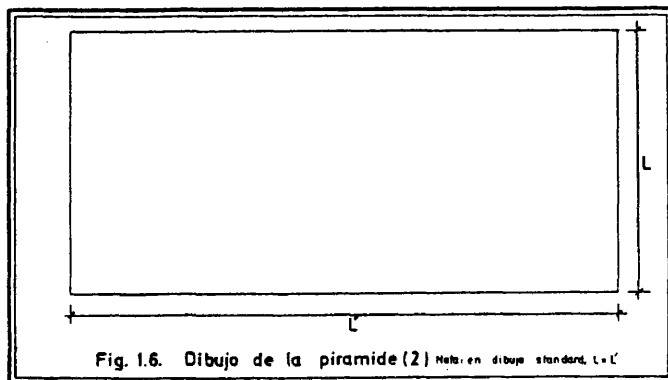
El software funcional del sistema Calcomp ofrece una subrutina fundamental que permite mover la pluma de dibujo, levantada -sin dibujar- o bajada -dibujando-, definiendo las coordenadas del punto al que debe dirigirse y una variable de control, IP -cuando el valor de esta variable es 2, la pluma se desplaza a ese punto dibujando; cuando el valor es 3, se desplaza sin dibujar-.

El problema reside, por tanto, en definir todas las coordenadas de los puntos que debe recorrer la pluma, pintando o sin pintar, para obtener el gráfico deseado.

#### 1.3.4.2.1. Dibujo de los ejes verticales.

Es un problema muy sencillo.

Partimos del cuadrado fundamental al que va a ajustarse el dibujo (Fig. 1.6.).



Consideramos el margen lateral que va a separar el dibujo de la pirámide (M) y la separación entre sus dos ramas (b), que hemos supuesto que es de la misma magnitud.

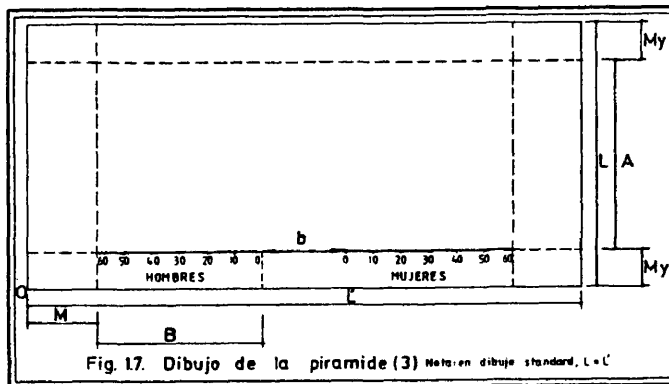
En función de estos parámetros se calcula la dimensión de los ejes horizontales (B) y, en función de ésta, la altura de la pirámide (A) (vid. supra).

Se calculan los márgenes superior e inferior del dibujo (My) en base a la siguiente ecuación

$$My = (L - A)/2$$



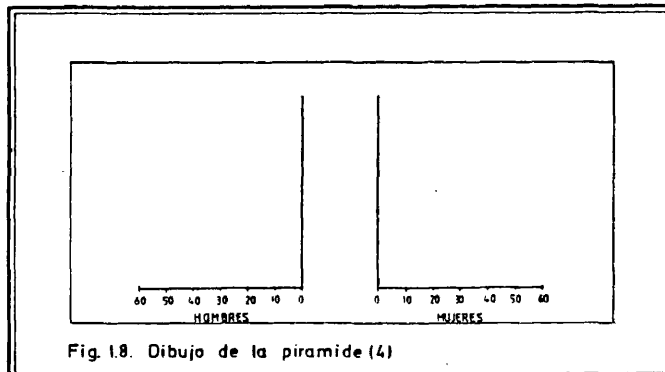
Los ejes horizontales se han dibujado (vid.supra) recurriendo a la utilización de las subrutinas SCALE y AXIS (Fig.1.7.).



Se sitúa el origen de coordenadas en el ángulo inferior izquierdo del cuadrado al que se va a ajustar el dibujo (O).

Teniendo en cuenta las definiciones anteriores, se comprende fácilmente que el problema de dibujar los ejes verticales se resuelve, para el eje femenino desplazando la pluma, sin pintar, hasta el punto de coordenadas  $(M+B+b, My+A)$  y moviéndola pintando hasta el de coordenadas  $(M+B+b, My)$ . Para el caso del eje masculino, el procedimiento es el mismo y los puntos en cuestión son respectivamente  $(M+B, My+A)$  y  $(M+B, My)$ .

El resultado gráfico hasta el momento presente es el siguiente ( Fig.1.8.).



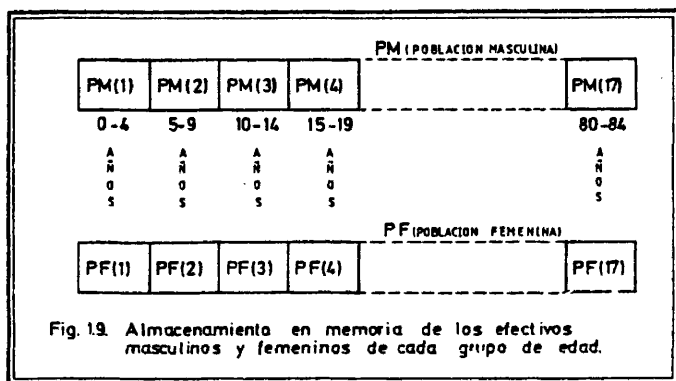
### 1.3.4.2.2. Dibujo de los escalones.

Esta parte del programa constituye un procedimiento iterativo que resuelve el dibujo de cada escalón utilizando el mismo conjunto de instrucciones.

Vamos a describir esquemáticamente, mediante un ejemplo, cómo se dibujan el escalón masculino y el femenino de un determinado grupo de edad.

Supongamos que vamos a pintar los escalones masculino y femenino del grupo de 0-4 años de una pirámide de población de grupos quinquenales de edad.

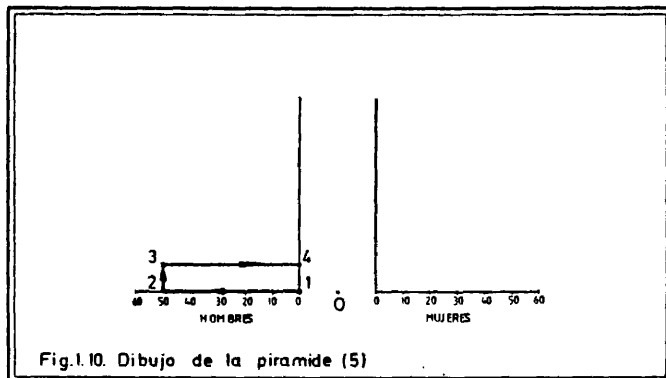
Para llevarlo a cabo el ordenador debe conocer, además de los parámetros fundamentales del dibujo, el valor exacto del efectivo masculino y del femenino de ese grupo de edad. Estos valores se encuentran almacenados en la memoria del ordenador en la primera posición de sendos vectores PM y PF, cada uno de los cuales tiene un número de elementos igual al de grupos de edad que se consideran, ordenados todos ellos de menor a mayor edad.



Desplazamos, en primer lugar, el origen de coordenadas al punto central de la base de la pirámide -esto se logra informando al ordenador de las coordenadas de ese punto respecto del anterior origen de coordenadas  $(M+B+b/2, My)$  y definiendo la variable de control como  $(-3)-$ .

A continuación se han de calcular todos los puntos que ha de recorrer la pluma para:

1. Dibujar el escalón masculino. (Fig. 1.10.).
2. Situar el rótulo conveniente de la escala vertical de la pirámide. (Fig.1.11.).
3. Dibujar el escalón femenino. (Fig. 1.12.).



Es necesario calcular las coordenadas de los puntos 1, 2, 3 y 4 del gráfico precedente, para dibujar el escalón masculino, y dirigir la pluma a 1 sin pintar, recorriendo los otros tres puntos pintando.

Las coordenadas de los puntos 1 y 4, teniendo en cuenta el significado de los siguientes parámetros, son fácilmente calculables:

$b$  = separación entre las dos ramas de la pirámide.

$a$  = altura de cada escalón (altura de la pirámide/nº de grupos de edad).

$$\text{punto 1 } (x_1, y_1) = (-b/2, a \cdot (\text{n}^\circ \text{ de orden del escalón} - 1))$$

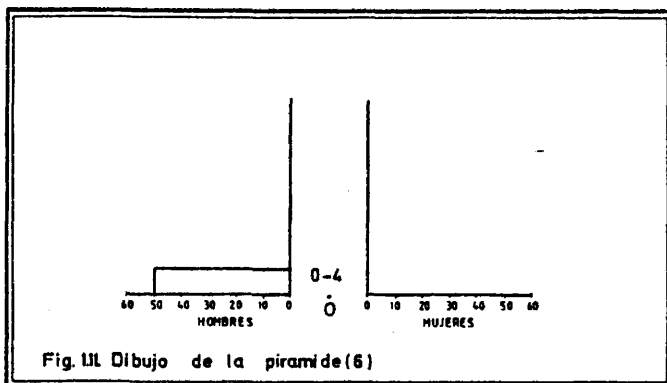
$$\text{punto 4 } (x_4, y_4) = (x_1, y_1 + a)$$

Para definir el valor de las coordenadas de los puntos 2 y 3, es necesario recordar ahora que mediante la utilización de la subrutina SCALE conocíamos el valor del factor escala ( $e$ ) de los ejes horizontales -número de unidades de medida de la población que se corresponden con 1 cm. del plano de dibujo-.

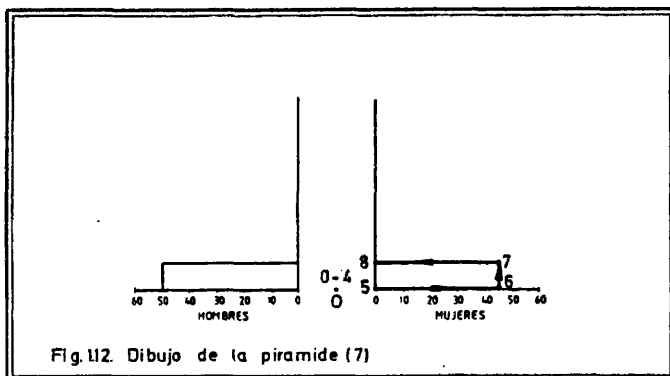
Teniendo en cuenta todas las notaciones precedentes se puede comprender que las coordenadas de esos puntos son las siguientes:

$$\text{Punto 2 } (x_2, y_2) = (-(b/2+PM(1)/e), y_1)$$

$$\text{Punto 3 } (x_3, y_3) = (x_2, y_4)$$



Utilizando las subrutinas NUMBER y SYMBOL del software funcional del sistema Calcomp, podemos rotular, a la altura deseada, el grupo de edad con que se corresponde el escalón pintado. Se trata de reproducir INF y SUP, dos parámetros de dibujo cuyo procedimiento de cálculo iterativo ya hemos descrito un poco más arriba.



Para reproducir ahora el escalón femenino es necesario desplazar, sin pintar, la pluma al punto 5 y recorrer 6, 7 y 8 pintando.

Atendiendo a la notación que hemos utilizado para definir los puntos correspondientes al escalón masculino, se pueden comprender, por analogía, los valores de las coordenadas de estos puntos:

$$\begin{aligned} \text{Punto 5 } (x_5, y_5) &= (b/2, y_1) \\ \text{Punto 6 } (x_6, y_6) &= (x_5 + PF(1)/e, y_1) \\ \text{Punto 7 } (x_7, y_7) &= (x_6, y_4) \\ \text{Punto 8 } (x_8, y_8) &= (x_5, y_4) \end{aligned}$$

Vemos, por tanto, que las únicas variables del problema, en el caso de que quisiéramos pintar un escalón cualquiera de la pirámide son: el valor del efectivo masculino o femenino de cada grupo de edad, los valores de las rotulaciones que hay que situar coherentemente a lo largo de los ejes verticales y la coordenada "y" de los dos puntos que definen la base de cada rectángulo de los que constituyen la pirámide. Pero resulta que estos valores, variables para el dibujo de cada escalón, son todos ellos función de otra magnitud, que es el n° de orden que corresponde a cada grupo de edad, una vez que éstos han sido ordenados de menor a mayor edad.

Sea  $i$  el n° de orden de un determinado grupo de edad;  $PM$  y  $PF$  dos vectores cuyos elementos son los efectivos masculinos y femeninos respectivamente en cada grupo de edad, ordenados de menor a mayor de edad;  $INF$  y  $SUP$  los rótulos correspondientes a cada escalón en la escala vertical;  $Y_B$  la coordenada "Y" de los puntos que constituyen la base de cada rectángulo.

Para todos los casos se cumple que:

Efectivo masculino de un determinado grupo de edad =  $PM_i$

Efectivo femenino de un determinado grupo de edad =  $PF_i$

$INF = (i-1) \cdot NA$        $NA =$  amplitud en años de los grupos de edad.

$SUP = (i \cdot NA) - 1$

$Y_B = a \cdot (i - 1)$        $a =$  altura de los escalones de la pirámide.

Cuando se produce una situación como la que acabamos de ver, en que un reducido número de instrucciones de cálculo o manipulación de datos afecta al conjunto de los miembros de una lista vectorial y que la naturaleza del resultado que pretende obtenerse de-

pende no sólo de los valores de la lista sino del orden en que éstos aparecen, se debe recurrir a la utilización de los procedimientos iterativos que ofrece cualquier lenguaje de programación\*.

Un procedimiento iterativo es un conjunto de instrucciones de manipulación de datos que se repite un cierto número de veces, afectando en cada caso a un conjunto de datos distinto. En FORTRAN\*, el procedimiento iterativo por excelencia es la utilización de la sentencia DO, aunque no es el único.

Solamente decir que para el caso concreto del dibujo de la pirámide es el procedimiento que se ha utilizado, ya que no voy a detenerme en la explicación de cómo funciona paso a paso un mecanismo de esta índole. Se puede recurrir a cualquiera de los manuales de programación en FORTRAN existentes, o, para las personas que tienen conocimientos básicos de programación, estudiar el listado de la subrutina, que aparece en el apéndice final.

#### 1.3.5. La realización automática de pirámides compuestas.

La pirámide de población compuesta se utiliza con frecuencia para representar gráficamente la distribución por edades y sexos de dos o más subconjuntos que se pueden distinguir dentro de una población, en base a un determinado criterio de clasificación.

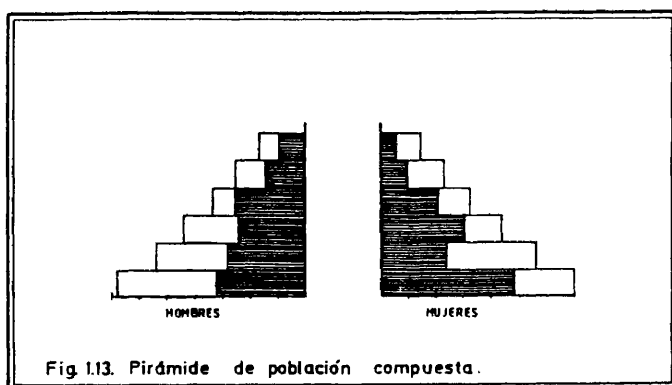
El procedimiento más usual es el de trazar los rectángulos que representan los efectivos masculinos y femeninos de una determinada edad, coherentemente con el diverso desarrollo de las clases definidas.

Lo más lógico, cuando se trata de representar pirámides compuestas, en función de un criterio interno de clasificación, es asignar la trama más densa al subgrupo de mayor interés y reservar la parte interior de cada rectángulo.

De esta manera se consigue una imagen doble que permite comparar gráficamente la estructura por edad y sexo del conjunto de la población (dibujo de la pirámide exterior de trazos rectos) y la del subgrupo que hemos considerado de mayor interés (pirámide interior de tramas) (Fig. 1.13.).

Si, por ejemplo, estamos interesados en el estudio de la variación de la actividad laboral de una determinada población en función de la edad y del sexo, lo más lógico es representar, en el interior de la pirámide del total de la población, la pirámide de la población activa, recurriendo a este procedimiento.

Este tratamiento sigue siendo válido, cuando pretendemos comparar la estructura por edad y sexo de más de un subgrupo de la población con la del total de la población, solamente en el caso de que exista una jerarquía en la clasificación, es decir cuando el nuevo subgrupo que se considera sea una partición del subconjunto que previamente se había considerado de mayor interés.



Me refiero a situaciones como las siguientes:

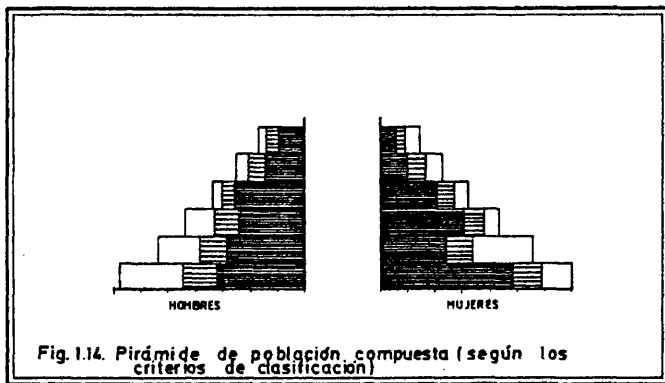
a. Representar la pirámide compuesta que representa la distribución de la población por edad y sexo del total de una provincia española, de su población urbana y de la población de su capital.

b. Obtener un gráfico de la distribución por edad y sexo de la población de una determinada ciudad, de su población activa, y de la población activa de alguno de sus distritos.

En ambos casos y en todos los supuestos semejantes a ellos, si se toma como medida del efectivo de un determinado grupo de edad, sexo y de otra o más características su porcentaje respecto del total de la población -para facilitar la comparación con otras situaciones-, y se toma siempre el mismo origen para representar los -la pirámide exterior, que corresponde a la población total, mediante rectángulos y las interiores, mediante tramas sucesivamente más densas- se puede obtener una representación como la de la figura 1.14.

En ella se pueden comparar simultáneamente las tres pirámides, ya que por tener todas las dimensiones su origen en el mismo eje, la forma de los bordes, que es el elemento básico de reconocimiento y análisis de las pirámides, es significativa.

La subrutina PIRAM2 ha sido diseñada para efectuar automáticamente este tipo de tratamientos gráficos.



No está, sin embargo, diseñada esta subrutina para representar en el interior de la pirámide de la población más de dos grupos que sean mutuamente excluyentes, por el hecho de que pienso que no es un sistema idóneo de representación, a pesar de que se recurra a él en algunas publicaciones de resultados de investigación.

Si se pretende, por ejemplo, representar dentro de la pirámide de población de un determinado colectivo la variación del estado civil en función de la edad y del sexo -que define tres situaciones distintas: solteros, casados y separados-, y se asigna a la porción correspondiente a cada clase una determinada trama, tomando como origen de cada clase, en cada escalón, el extremo de la anterior, el dibujo de los bordes de las zonas de tramado semejante no es significativo, ya que su origen no es común en todos los casos.

En mi opinión este tipo de representación es un intento fallido de sintetizar un mayor volumen de información en un único gráfico. Veo, sin embargo, mucho más lógico el obtener tres gráficos distintos -en el momento presente no plantea problema un aumento en el número de gráficos a obtener, ya que éstos se realizan automáticamente con un costo mínimo de tiempo y esfuerzo- en cada uno de los cuales se representa en el interior de la pirámide general, la pirámide de edad del grupo correspondiente -en el caso del ejemplo propuesto, la de los solteros, la de los casados y la de los separados-.

La subrutina PIRAM2 debe conocer los mismos parámetros fundamenta



les de dibujo que la PIRAM, por este motivo se ha reservado un espacio común de memoria compartido por ambas.

Recordando lo que hemos señalado más arriba, estos parámetros son los siguientes:

- La escala horizontal del dibujo, es decir, la proporción que existe entre la unidad lineal del plano de representación y la unidad de medida de la magnitud de la población. (e)

- Margen lateral que separa el dibujo de la pirámide del cuadro general en que se enmarca. (M)

- Separación entre las dos ramas de la pirámide. (b)

- Altura de los escalones correspondientes a cada grupo de edad. (a)

- Longitud de la base de cada una de las ramas de la pirámide. (B)

Por otra parte, la trama que efectúa esta subrutina es un rayado paralelo al eje de abscisas de los rectángulos ya dibujados por PIRAM. La densidad de la trama es, lógicamente, proporcional al número de rayas que se van a trazar en el interior de cada rectángulo.

Vamos a ver someramente cómo funciona la subrutina para el caso de una pirámide compuesta, en función de un criterio de clasificación, de dos clases resultantes por cada grupo de edad y sexo.

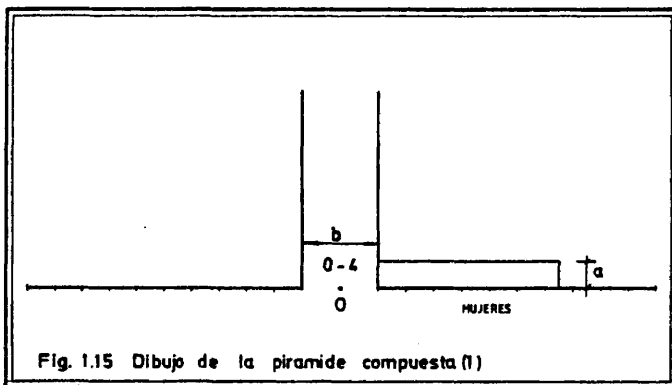
Nos limitamos a un único escalón de la pirámide, ya que el procedimiento es iterativo y exactamente igual en los demás casos.

El resultado del funcionamiento de PIRAM en un paso anterior, ha sido el dibujo de la figura 1.15.

Conociendo el valor de la población femenina de edad 0 - 4 años y de otra determinada característica -supongamos que la pirámide exterior corresponde a la provincia de Madrid y la interior a la de Madrid capital- en un porcentaje de la población total, PIRAM2 dibuja en primer lugar una barra vertical que define la extensión gráfica del nuevo grupo de edad (Fig. 1.16.).

Para ello es necesario calcular las coordenadas de los puntos P1 y P2, en función de los datos ya conocidos, desplazar la pluma sin pintar al punto P1 y desplazarla a continuación a P2 pintando.

Las coordenadas de los dos puntos se pueden calcular en base a las siguientes expresiones:



/1.1/  $X_{p1} = X_{p2} = b/2 + PF_i/e$      $PF =$  vector de valores de todos los efectivos femeninos de una determinada característica, ordenados por grupos de edad creciente

$i = n^\circ$  de orden

/1.2/  $Y_{p1} = a \cdot (i - 1)$

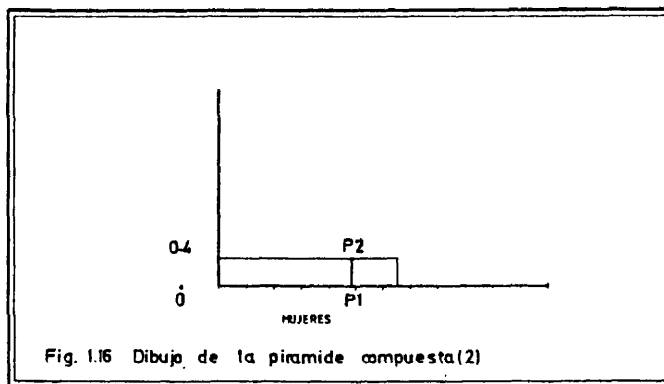
/1.3/  $Y_{p2} = a \cdot i$

A continuación traza un haz de paralelas que se sitúa en el interior del nuevo rectángulo así definido. La trama standard sería ocho líneas paralelas en el interior del polígono, pero para facilitar la comprensión del procedimiento supongamos que sólo se van a trazar dos (Fig. 1.17.).

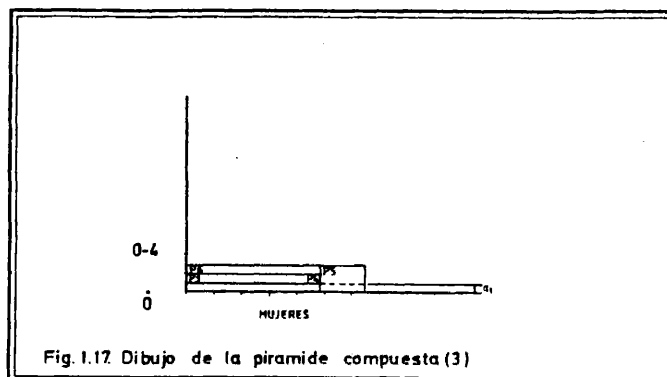
El problema se reduce a calcular las coordenadas de los puntos P3, P4, P5 y P6 y dirigir posteriormente la pluma sin dibujar a P3, desplazandola a P4 dibujando, moverla a P5 sin dibujar y finalmente llevarla a P6 dibujando.

Las coordenadas de estos puntos se pueden calcular en base a las siguientes ecuaciones.

/1.4/  $X_{p3} = X_{p6} = b/2$



- /1.5/  $X_{p4} = X_{p5} = b/2 + PF_1/e$      $a_1 = a/(n^\circ \text{ de rayas} + 1)$   
 /1.6/  $Y_{p3} = Y_{p4} = a_1 + a_1 \cdot (j-1)$      $j = n^\circ \text{ de orden de la raya}$   
 /1.7/  $Y_{p5} = Y_{p6} = a_1 + a_1 \cdot j$



Utilizándola convenientemente, la misma subrutina sirve para hacer un nuevo tramado, cuyo resultado es obtener una trama más densa en el interior de la pirámide, coherente con el desarrollo de la población de una determinada edad y sexo y de más de una característica adicional.

Aunque el programa general puede acometer la tarea de representar pirámides compuestas atendiendo a más de dos criterios, sólo se han hecho pruebas de representar situaciones en las que se considere un máximo de dos criterios de clasificación. Por otra parte resulta difícil pensar que sean realmente legibles gráficos de mayor complejidad. De cualquier forma ahí está la posibilidad de experimentar con ellos, ya que la obtención de los mismos no supone en ningún caso un esfuerzo estimable.

Consideramos la situación en la que se pretende obtener una segunda trama interior a la primera, cuyo desarrollo total corresponde con el de un efectivo determinado de población caracterizado por su edad, sexo y dos características más -siguiendo el ejemplo que hemos citado anteriormente, se trata de representar la población femenina de 0 a 4 años de un determinado distrito de la ciudad de Madrid, en el interior de la correspondiente a Madrid ciudad y a su vez en el interior de la correspondiente a la provincia-.

Hasta ahora, por la utilización combinada de las subrutinas, PIRAM -en base a los datos provinciales- y PIRAM2 -en función de los datos de la capital-, habíamos obtenido el siguiente gráfico (Fig. 1.18.).

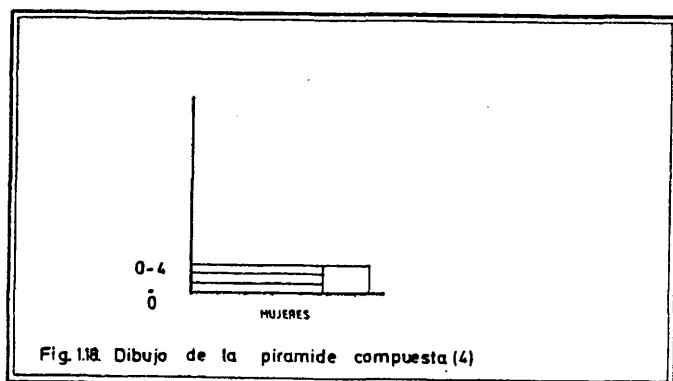


Fig.1.18. Dibujo de la piramide compuesta (4)

Si, con referencia a los datos del nuevo grupo de individuos clasificado también por edad y sexo, utilizamos otra vez la subrutina PIRAM2, se comprende fácilmente que el resultado de la primera fase de ésta dé como resultado la figura 1.19.

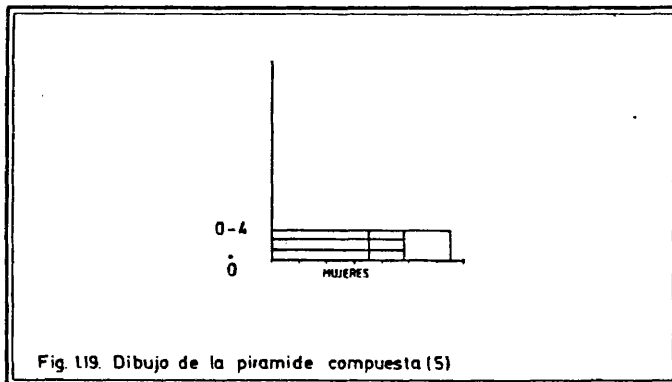


Fig. 1.19. Dibujo de la piramide compuesta (5)

El tramado consecuente, que se realiza después, cuenta ya con la trama previa y sitúa nuevas líneas intercaladas con las anteriores, produciendo una trama de densidad doble en el interior de la pirámide (Fig. 1.20.).

Aparentemente esta segunda fase del dibujo parece diferir de la que realizaba PIRAM2 en el ejemplo precedente, pero no es así. La subrutina de dibujo es exactamente la misma.

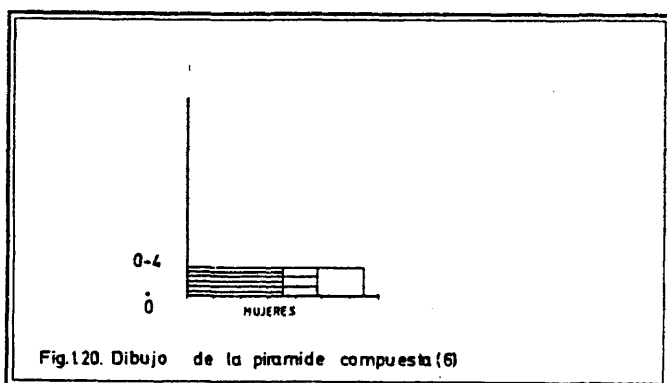
Estableciendo dos precisiones, se podrá comprender.

En primer lugar, aunque no hayamos hecho alusión hasta ahora, para no complicar innecesariamente las cosas, las ecuaciones /1.6/ y /1.7/ las conoce el ordenador en una versión que difiere un poco. Las ecuaciones reales que permiten calcular las coordenadas de los puntos necesarios para establecer el tramado son las siguientes:

$$/1.8/ \quad Y_{P3} = Y_{P4} = a_1 + a_1 \cdot (j-1) - a_1 \cdot d$$

$$/1.9/ \quad Y_{P5} = Y_{P6} = a_1 + a_1 \cdot j - a_1 \cdot d$$

Siendo  $d$  un valor que se hace 0.0 la primera vez que se utiliza PIRAM2 y 0.5 en la segunda ocasión.



En segundo lugar observamos que en este segundo caso se ha dibujado una raya más, pero no es difícil automatizar un proceso mediante el cual cada vez que se utilice de nuevo PIRAM2 se obtenga un número de rayas superior en una a las que ya se han producido en la suma de los tramados anteriores. Piénsese que para lograr el mismo efecto, en el caso de una nueva trama, sería necesario producir 6 líneas intercaladas en las anteriormente dibujadas, y así sucesivamente.

#### 1.3.6.- La superposición de dos pirámides de población.

Otro tratamiento gráfico que puede resultar de interés es la superposición de dos pirámides de población de dos grupos distintos, para establecer una comparación de sus diferentes estructuras.

Piénsese, por ejemplo, en la expresividad de la superposición de las pirámides de población de diversos distritos de una

ciudad sobre la pirámide del conjunto de la misma, que actúa como elemento de comparación de las divergencias que se verifican respecto a ella.

El mismo tipo de planteamiento subyace en la pretensión de representar cualquier estructura de edad en comparación con la estructura de una población tipo -definida previamente en función de determinadas pautas demográficas-.

La solución de este problema puede acometerse por dos caminos diferentes que intentan ambos un mismo objeto: distinguir perfectamente cada una de las dos pirámides superpuestas.

El camino más sencillo es diseñar una subrutina muy próxima a PIRAM -con algunas diferencias que no merece la pena comentar-, que utilice una pluma de color distinto, para pintar la segunda pirámide -la que se va a dibujar sobre la población de referencia-.

Esta subrutina ya ha sido escrita y comprobada, pero no la presento en este trabajo porque pienso que no es una solución adecuada, y esto por un motivo fundamental, los problemas que acarrean las tiradas en imprenta a dos o más colores.

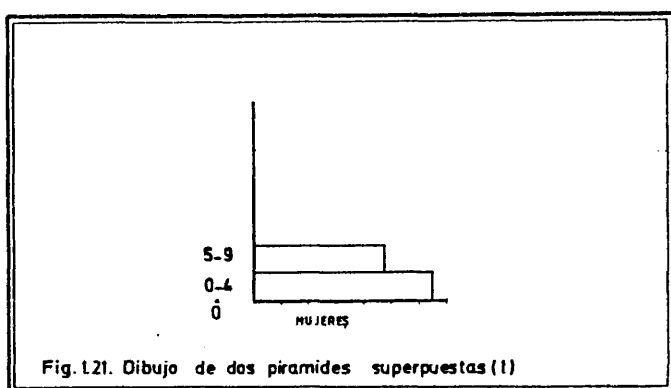
Aunque resulte muy sencillo la obtención de gráficos a varios colores mediante ordenador, no hay que dejarse deslumbrar por esta facilidad, ni por la expresividad de los dibujos a más de una tinta, ya que luego esto encarece tremendamente su reproducción en imprenta -además muchas publicaciones científicas no admiten más que trabajos en blanco y negro-. Por este motivo, no hay que recurrir al color, más que cuando no haya otro medio viable de presentar una determinada realidad.

En el caso presente, aunque el trabajo de programación y el costo de ordenador se incrementa, es perfectamente válido solucionar el problema a una sola tinta, recurriendo a dos tramas expresivas, la una, del mayor desarrollo del escalón correspondiente a la pirámide de referencia, y la otra de la situación opuesta. Más concretamente me ha parecido lógico asignar la trama más densa a la segunda situación y la menos densa a la primera.

El procedimiento a seguir es el que comento a continuación, cambiando, como he hecho hasta ahora, en el dibujo de un número reducido de escalones, para facilitar su comprensión.

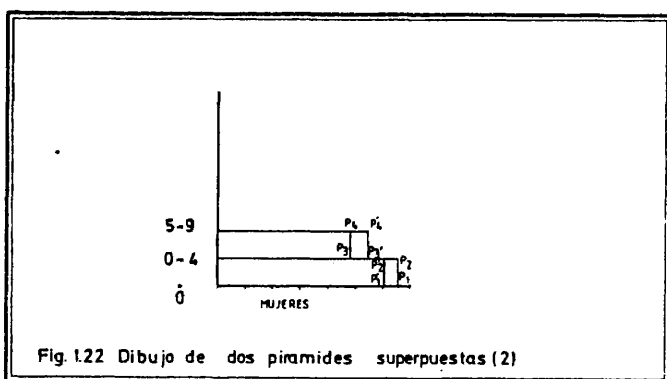
En primer lugar, recurriendo a PIRAM se dibuja la pirámide de edad de la población de referencia (Fig. 1.21.).

A continuación, y en la misma tinta, se dibuja sobre el mismo marco la pirámide a comparar, utilizando PIRAM pero, en este caso concreto, saltándonos el paso donde se calcula la escala horizontal, y asumiendo la que ya había sido calculada por PIRAM en su primera utilización.



Además, las poblaciones de cada grupo de edad y sexo se presentan como porcentaje respecto del total de su población -no como porcentajes de la población de referencia, de la que la segunda, puede, pero perfectamente puede no ser una parte-.

El resultado gráfico supongamos que sea el siguiente (Fig. 1.22.).





Se han formado dos paralelepípedos, que ahora es necesario tramar convenientemente en función del objetivo deseado. El polígono definido por  $(P_1, P_2, P'_2, P'_1)$  será tramado, tal como hemos decidido anteriormente por una trama menos densa -supone un predominio del escalón de la pirámide de referencia- que el polígono  $(P'_3, P'_4, P_4, P_3)$ .

Llegados a este punto es necesario hacer un salto en el vacío para referirnos a algo que se explica más detenidamente en páginas posteriores de la tesis, pero que puede comprenderse en su aspecto general.

Trabajando con medios informáticos es mucho más coherente intentar resolver problemas de tipo general, de los que muchos objetivos de cálculo o manipulación de datos sean una particularización de los mismos, que considerar cada tarea de programación como un problema en sí a resolver.

En este sentido, para tramar estos paralelepípedos podía haber recurrido a diseñar una nueva subrutina de rayado -probablemente utilizando parte del texto de PIRAM2- para resolver el problema.

Pero mucho más adecuado es redactar una subrutina general de tramado de cualquier polígono, que solamente necesite conocer el polígono a tramar (definido por sus vértices) y una información adicional, que especifica el tipo de trama elegido de las de un repertorio que ofrece la subrutina.

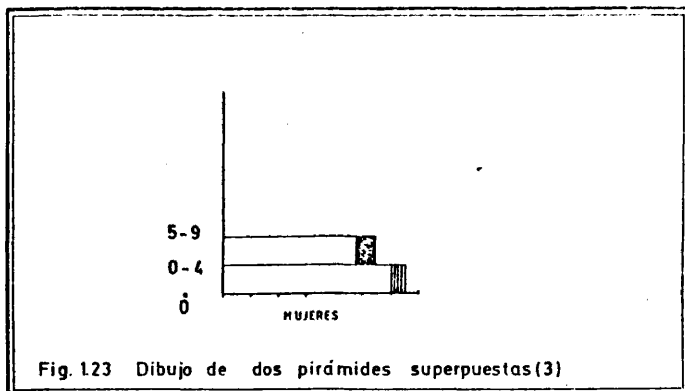
La descripción detallada de esta subrutina de tramado puede encontrarse en el capítulo segundo.

Teniendo en cuenta todo esto, el procedimiento de resolución gráfica de pirámides superpuestas sería de la siguiente índole.

Al pintar la primera pirámide de población, almacenar en memoria las coordenadas de los puntos de los extremos de los escalones.

Efectuar la misma operación al dibujar la segunda pirámide, almacenando los extremos de los escalones de forma coherente con el anterior procedimiento.

Finalmente, utilizar la subrutina TRAMA para rellenar todos los polígonos definidos, investigando antes si se trata de un caso de predominio de la pirámide de referencia, o a la inversa, para reproducir la trama que sea conveniente (Fig. 1.23.).



### 1.3.7. Información adicional, en función de las necesidades del usuario.

En su versión standard las subrutinas a que hemos hecho alusión no aportan ninguna información adicional. Sólomente, cuando se desean dibujar varias pirámides dentro de un único trabajo, se sitúa en el ángulo superior izquierdo del marco que encuadra cada pirámide un identificador numérico del orden en que ha sido dibujada.

Sin pretender agotar todas las posibilidades, ya que pienso que las tareas de rotulado de dibujos son suficientemente sencillas como para poder ir siendo desarrolladas por alumnos de la especialidad con conocimientos informáticos, he diseñado el mínimo imprescindible de ellas para poder ofrecer al usuario de estas rutinas de dibujo unos rótulos suficientemente expresivos.

En primer lugar, si lo desea el usuario, puede obtener un rótulo de hasta 77 caracteres -incluidos los espacios en blanco- que se sitúa en la base de la pirámide, para identificar qué realidad está representando (Fig. 1.24.).

También, para el caso de pirámides compuestas, o de superposición de pirámides, puede obtener en el margen derecho una leyenda gráfica de las tramas (Fig. 1.25.).

Finalmente, si el usuario lo desea e indica el año de referencia de la pirámide, se sitúa en el borde derecho de la pirámide de una escala con los años de nacimiento de los individuos de la base de cada grupo de edad (Fig. 1.26.).

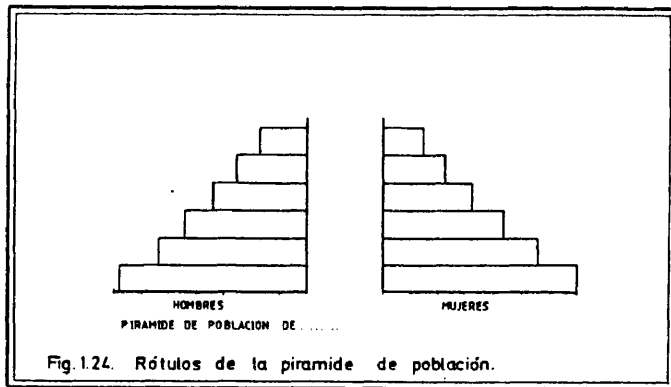


Fig.1.24. Rótulos de la piramide de población.

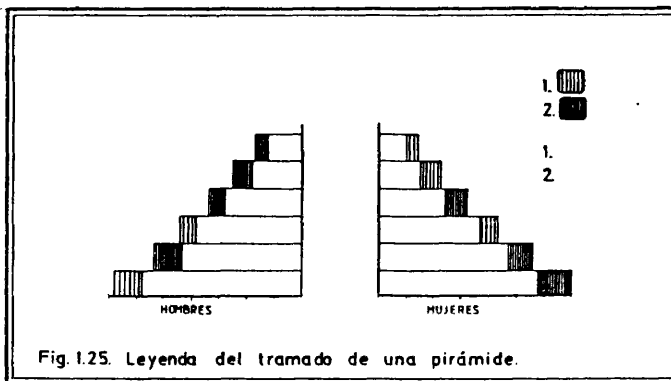
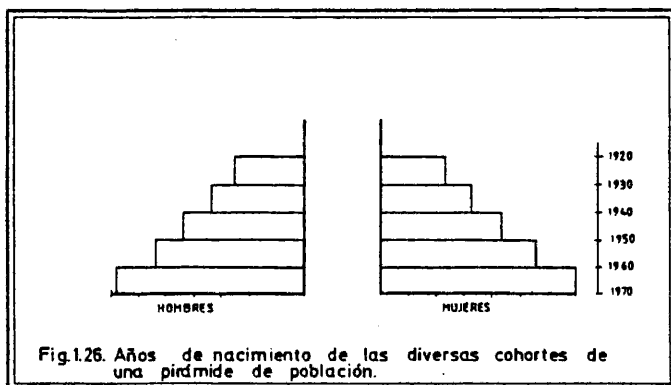


Fig.1.25. Leyenda del tramado de una pirámide.



#### 1.3.8. Flexibilidad de los parámetros fundamentales de dibujo.

Como ya comentábamos unas páginas más arriba, experimentando el dibujo de pirámides con diversos conjuntos de datos -con ocasión de la presentación de investigaciones concretas (Memorias de Licenciatura), que han utilizado estos procedimientos- se ha visto conveniente permitir al usuario una mayor libertad de movimientos en lo que respecta a la definición de los parámetros fundamentales del dibujo.

Si el usuario no opta por modificarlos, todos los parámetros se calcularán en base a las dimensiones del cuadrado general que enmarca el dibujo de la pirámide.

Por la experiencia de utilización del programa en su versión standard se ha visto necesario facilitar la definición optativa del marco del dibujo, de la escala horizontal del dibujo (nº de unidades de medida de los efectivos de la población/cm.), de la base de la pirámide y de la altura de la misma, así como de todos los márgenes del dibujo y de la separación de las ramas de la pirámide.

### 1.3.9. Requisitos y organización de los datos para uso del programa.

En el listado del programa que presentamos a continuación aparece, como en todos los demás casos, una documentación exhaustiva acerca de los requisitos y organización de los datos de partida.

Vamos ahora a subrayar, tan sólo, la estructura básica de la información que manipula el programa, que consta de dos bloques o ficheros\*.

El primer fichero lo constituyen los datos de estructura por edad y sexo a representar y los parámetros que describen la naturaleza de éstos: formato con el que han sido grabados, número de grupos de edad, amplitud en años de los mismos y ámbito al que corresponden.

En el segundo fichero se definen las transformaciones que se desean realizar con los datos antes de representarlos, las modificaciones de los parámetros standard de dibujo que se solicitan y los optativos de información gráfica complementaria.

Aunque en la documentación únicamente se haga referencia a tarjetas perforadas, el soporte informático de entrada de datos puede ser cualquiera, siempre que la definición de los ficheros, mediante sentencias de JCL\*, sea coherente con el soporte utilizado (cinta, disco, etc.).

Finalmente, a pesar de que el usuario del programa tenga en sus manos una amplia gama de mecanismos de control, es imprescindible que los datos de estructura por edad y sexo estén grabados, para cada pirámide a representar, de la siguiente manera: efectivos masculinos ordenados de menor a mayor edad, efectivos femeninos ordenados de la misma forma. Si los datos del usuario no tienen esta estructura, antes de modificar el programa, compensa diseñar una subrutina simple de preparación coherente de la información.

### 1.3.10. Listado del programa

Como en todos los demás casos, presentamos al final de cada apartado el listado del programa que realiza las tareas descritas previamente.

Dado que todos los programas utilizan un cierto número de subrutinas, y que éstas, a veces, son compartidas por varios programas a la vez, he preferido presentar todas ellas al final, en un apéndice, por orden alfabético, para facilitar su consulta.

Tanto los programas principales como las subrutinas están comentados en sus partes fundamentales, para facilitar su comprensión.

si6n a personas con conocimientos inform6ticos.

Por 6ltimo, decir, y esto es v6lido para todos los programas que presento, que as6 como las subrutinas son en su mayor6a productos acabados -aunque, por supuesto, susceptibles de mejoras-, los programas no son m6s que una cierta combinaci6n de subrutinas, que ha resultado de inter6s hasta el momento. Constituyen, por tanto, un estado de desarrollo dentro de un proceso de enriquecimiento y mejora, aunque lo suficiente pr6ximo a un 6ptimo aceptable como para ser presentados a unos futuros usuarios de los mismos que, en la mayor6a de los casos, no necesitar6n m6s que un m6nimo de conocimientos inform6ticos para beneficiarse de sus prestaciones.



C SI A = 2, ESTOS DATOS SE LEEN CON EL FORMATO DESCRITO POR EL USUA-  
 C RIO.  
 C SI B = 1, EL PROGRAMA INTERPRETA QUE EL FORMATO DE LECTURA, EL NU-  
 C MERO DE GRUPOS DE EDAD Y LA AMPLITUD DE LOS MISMOS SON IGUALES PA-  
 C RA TODAS LAS PIRAMIDES.  
 C SI B = 2, EL PROGRAMA INTERPRETA QUE EL FORMATO DE LECTURA, EL NU-  
 C MERO DE GRUPOS DE EDAD Y LA AMPLITUD DE LOS MISMOS SON DISTINTOS  
 C PARA CADA PIRAMIDE.  
 C  
 C TERCERA TARJETA: SE DESCRIBE EL FORMATO DE LOS DATOS, CUANDO NO ES  
 C EL STANDARD.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 C 1-EL FORMATO DE LOS DATOS ES(.....)  
 C  
 C SI A = 1, ESTA TARJETA NO DEBE APARECER.  
 C SI B = 1, APARECE SOLAMENTE UNA VEZ. SI B = 2, APARECE TANTAS VECES  
 C COMO PIRAMIDES DE POBLACION, DELANTE DE LOS DATOS QUE DESCRIBE.  
 C  
 C CUARTA TARJETA: SE DESCRIBE EL NUMERO DE GRUPOS DE EDAD.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 C 1-EL NUMERO DE GRUPOS DE EDAD ES --  
 C  
 C SI B = 1, APARECE SOLAMENTE UNA VEZ. SI B = 2, APARECE TANTAS VECES  
 C COMO PIRAMIDES DE POBLACION, DELANTE DE LOS DATOS QUE DESCRIBE.  
 C  
 C QUINTA TARJETA: SE DESCRIBE LA AMPLITUD DE LOS GRUPOS DE EDAD.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 C 1-LOS GRUPOS SON DE -- AÑOS  
 C  
 C SI B = 1, APARECE SOLAMENTE UNA VEZ. SI B = 2, APARECE TANTAS VECES  
 C COMO PIRAMIDES DE POBLACION, DELANTE DE LOS DATOS QUE DESCRIBE.  
 C  
 C SEXTA TARJETA, QUE DEBE APARECER SIEMPRE DELANTE DE LOS DATOS DE  
 C CADA PIRAMIDE: SE DEFINE EL TIPO DE PIRAMIDE Y, SI SE DESEA, UN RO-  
 C TULO EXPRESIVO DE HASTA 77 CARACTERES, QUE SE REPRODUZCA EN EL DI-  
 C BUJO.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 C 1-PIRAMIDE DE POBLACION DE.....  
 C  
 C SI C = 1, SE TRATA DE UNA PIRAMIDE SIMPLE.  
 C SI C = 2, SE TRATA DE UNA PIRAMIDE INTERIOR A OTRA.  
 C SI C = 3, SE TRATA DE UNA PIRAMIDE A SUPERPONER SOBRE OTRA.  
 C  
 C LECTURA DE LOS EFECTIVOS MASCULINOS Y FEMENINOS CON FORMATO STAN-  
 C DARD.  
 C  
 C POBLACIONES MASCULINAS( SOBRE LAS TARJETAS QUE SEAN NECESARIAS).  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 C 7546 6534 5420.....987  
 C  
 C



```

C  PDBLACIONES FEMENINAS( SOBRE LAS TARJETAS QUE SEAN NECESARIAS).
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C      7235      6421      4920.....1087
C
C  LECTURA DE LOS EFECTIVOS MASCULINOS Y FEMENINOS CON FORMATO DEFINI-
C  DO POR EL USUARIO (POR EJEMPLO:(10F8.0)).
C
C  POBLACIONES MASCULINAS( SOBRE LAS TARJETAS QUE SEAN NECESARIAS).
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C      7546      6534      5420.....987
C
C  POBLACIONES FEMENINAS( SOBRE LAS TARJETAS QUE SEAN NECESARIAS).
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C      7235      6421      4920.....1087
C
C  ULTIMA TARJETA.
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C  /*
C
C
C  FICHERO FT07F001
C  *****
C
C  PRIMERA TARJETA.
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C  //FT07F001 DD *
C
C  SEGUNDA TARJETA: SE DEFINEN LAS TRANSFORMACIONES QUE SE SOLICITAN.
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C  2-TRANSFORMACIONES D      E
C
C  SI D = 1, SE EFECTUA UNA AGREGACION QUINQUENAL (PROCEDIMIENTO STAN-
C  DARD).
C  SI D = 2, SE EFECTUA LA AGREGACION QUE DESEE EL USUARIO CON LOS DA-
C  TOS DE TODAS LAS PIRAMIDES.
C  SI D = 3, EL USUARIO DESEA UN TIPO DE AGREGACION DISTINTO, QUE DEBE
C  EXPLICITAR EN TODOS LOS CASOS, PARA CADA PIRAMIDE.
C  SI E = 1, LOS DATOS INICIALES SE TRANSFORMAN EN PORCENTAJES.
C  SI E = 2, SE PUEDE SEGUIR UNA DE ESTAS DOS OPCIONES: NO TRANSFORMAR
C  LOS DATOS O TRANSFORMARLOS DE ALGUNA MANERA QUE NO SEA EL CALCULO
C  DE PORCENTAJES (PARA ESTO ULTIMO EL USUARIO DEBE DEFINIR EN FORTRAN
C  LAS TRANSFORMACIONES QUE DESEA).
C
C  TERCERA TARJETA: SE DESCRIBE EL TIPO DE AGREGACION QUE SE SOLICITA.
C  1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C  2-LOS GRUPOS DE EDAD HAN DE SER DE -- A"DS
C
C  SI D = 1, ESTA TARJETA NO DEBE APARECER.
C  SI D = 2, ESTA TARJETA APARECE UNA VEZ.
C  SI D = 3, ESTA TARJETA APARECE TANTAS VECES COMO PIRAMIDES.

```

C  
C CUARTA TARJETA: SE DEFINEN EN TERMINOS GENERALES LOS MODIFICADORES  
C DE PARAMETROS.  
C 1 10 20 30 40 50 70 80  
COL. |.....|.....|.....|.....|.....|.....|.....|.....|  
C 3-MODIFICACION DE LOS PARAMETROS DE DIBUJO F

C  
C SI F = 1, DIBUJO STANDARD.  
C SI F = 2, MODIFICACION DE LOS PARAMETROS DE DIBUJO COMUN A TODAS LAS  
C PIRAMIDES.  
C SI F = 3, MODIFICACION DE LOS PARAMETROS DE DIBUJO ESPECIFICA PARA  
C CADA PIRAMIDE.

C  
C QUINTA TARJETA: NUEVOS VALORES DE LOS PARAMETROS(FORMATO(5F10.0)).  
C 1 10 20 30 40 50 70 80  
COL. |.....|.....|.....|.....|.....|.....|.....|.....|  
C GGGGGGGG HMMMMHHH IIIIIIII JJJJJJJJ KKKKKKKK

C  
C G = FACTOR DE ESCALA DEL DIBUJO.  
C H = ESCALA DE LOS EJES HORIZONTALES DEL DIBUJO (EN UNIDADES DE MEDI-  
C DA DE LA POBLACION/CM DEL PLANO DE DIBUJO).  
C I = ALTURA EN CM. DE LOS ESCALONES DE LA PIRAMIDE.  
C J = MARGEN EN CM. DEL DIBUJO RESPECTO DEL MARCO.  
C K = VARIABLE QUE, EN CASO DE TENER UN VALOR DISTINTO DE CERO, HACE  
C QUE LA EXTENSION EN LA HORIZONTAL DE LOS ESCALONES SE DIVIDA  
C POR LA AMPLITUD DE LOS GRUPOS DE EDAD.

C  
C SI F = 1, ESTA TARJETA NO DEBE APARECER.  
C SI F = 2, ESTA TARJETA APARECE UNA VEZ.  
C SI F = 3, ESTA TARJETA APARECE TANTAS VECES COMO PIRAMIDES.

C  
C SEXTA TARJETA: SE DESCRIBEN EN TERMINOS GENERALES LOS OPTATIVOS QUE  
C SE SOLICITAN.  
C 1 10 20 30 40 50 70 80  
COL. |.....|.....|.....|.....|.....|.....|.....|.....|  
C 4-OPTATIVOS L

C  
C SI L = 1, NO SE SOLICITA NINGUN OPTATIVO.  
C SI L = 2, SE SOLICITAN LOS MISMOS OPTATIVOS PARA TODAS LAS PIRAMIDES  
C SI L = 3, SE SOLICITAN OPTATIVOS DISTINTOS PARA CADA PIRAMIDE.

C  
C SEPTIMA TARJETA: SE EXPRESAN LOS OPTATIVOS ELEGIDOS.  
C 1 10 20 30 40 50 70 80  
COL. |.....|.....|.....|.....|.....|.....|.....|.....|  
C M N O P 0000

C  
C SI M = 1, SE REPRODUCE EL ROTULO INTRODUCIDO POR CADA PIRAMIDE.  
C SI N = 1, APARECE LA LEYENDA DE PIRAMIDES COMPUESTAS (UN CRITERIO).  
C SI N = 2, APARECE LA LEYENDA DE PIRAMIDES COMPUESTAS (DOS CRITERIOS)  
C SI O = 1, APARECE LA LEYENDA DE PIRAMIDES SUPERPUSTAS.  
C SI P = 1, APARECE LA LEYENDA DE LOS AÑOS DE NACIMIENTO DE LAS CO-  
C HORTES.  
C 0000 = FECHA CORRESPONDIENTE A LA PIRAMIDE.

C  
C SI L = 1, ESTA TARJETA NO DEBE APARECER.  
C SI L = 2, ESTA TARJETA APARECE UNA VEZ.  
C SI L = 3, ESTA TARJETA APARECE TANTAS VECES COMO PIRAMIDES.  
C  
C

```

C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
C COL.1.....|.....|.....|.....|.....|.....|.....|
C /*
C*****
C*****
C*****
C
COMMON/A/DATOS(90,40),ICOD(20),NGE(20),IAG(20),IAGF(20)
COMMON/B/ROTUL(80,21)
INTEGER OUT
CHARACTER ROTUL,FMT(80)
C
C DATOS = MATRIZ DONDE VAN A SER ALMACENADOS LOS DATOS DE ESTRUCTURA
C POR EDAD Y SEXO QUE SUMINISTRA EL USUARIO.
C ICOD = VECTOR ENTERO DONDE VAN A ALMACENARSE LOS CODIGOS QUE IDENTIFICAN LA NATURALEZA DE LOS DATOS. SI ICOD(I)=1, LAS DOS COLUMNAS CORRESPONDIENTES REPRESENTAN UNA NUEVA PIRAMIDE. SI ICOD(I)=2, SE TRATA DE DATOS CORRESPONDIENTES A UNA PIRAMIDE COMPUESTA. SI ICOD(I)=3, SE CONSIDERA QUE SE VAN A SUPERPONER DOS PIRAMIDES.
C NGE = VECTOR ENTERO DONDE SE ALMACENA EL NUMERO DE GRUPOS DE EDAD QUE TIENE CADA ESTRUCTURA A REPRESENTAR.
C IAG = VECTOR ENTERO DONDE SE ALMACENAN LOS VALORES DEL GRADO DE AGREGACION POR EDAD, QUE TIENEN LOS DATOS SUMINISTRADOS POR EL USUARIO.
C IAGF = VECTOR ENTERO DONDE SE ALMACENAN LOS VALORES DEL GRADO DE AGREGACION POR EDAD, DE LOS DATOS, QUE EL USUARIO DESEA EN LA REPRESENTACION.
C ROTUL = MATRIZ DE CARACTERES DONDE SE ALMACENAN TANTOS ROTULOS EXPLICATIVOS COMO PIRAMIDES A REPRESENTAR. ESTOS ROTULOS PUEDEN TENER UN MAXIMO DE 77 CARACTERES.
C FMT = VECTOR DE CARACTERES QUE SE UTILIZA PARA DESCRIBIR EL FORMATO DE LOS DATOS QUE APORTA EL USUARIO, SI ESTE NO COINCIDE CON EL STANDARD.
C
C*****
C
C INICIALIZACION DE LAS VARIABLES DE LECTURA Y ESCRITURA.
C
C DATA IN,OUT/5,6/
C
C TODA LA INFORMACION REFERENTE A LAS ESTRUCTURAS POR EDAD Y SEXO A REPRESENTAR GRAFICAMENTE CONSTITUYE EL FICHERO FTOSF001.
C
C*****
C
C SE LEEN LOS PARAMETROS DE LECTURA DE DATOS.
C
C READ(IN,100) (ROTUL(1,21),I=1,7),IFMT1,IFMT2
C 100 FORMAT(7A1,I3,T18,I3)
C WRITE(OUT,101) (ROTUL(1,21),I=1,7),IFMT1,IFMT2
C 101 FORMAT(1X,7A1,I3,T19,I3)
C
C*****
C
C LECTURA DE LOS DATOS.

```

```

C
  N=0
  1 CONTINUE
  N=N+1
  N2=N*2
  N2M1=N2-1
  IF(IFMT1.E0.1) GOTO 3
  2 CONTINUE
  IF(IFMT2.E0.1.AND.N.NE.1) GOTO 3
  READ(IN,102,END=10) (ROTUL(I,21),I=1,28),(FMT(I),I=1,52)
102 FORMAT(80A1)
  WRITE(OUT,103) (ROTUL(I,21),I=1,28),(FMT(I),I=1,52)
103 FORMAT(1X,132A1)
  3 CONTINUE
  IF(IFMT2.E0.1.AND.N.NE.1) GOTO 5
  READ(IN,104,END=10) (ROTUL(I,21),I=1,33),NGE(N)
104 FORMAT(33A1,12)
  WRITE(OUT,105) (ROTUL(I,21),I=1,33),NGE(N)
105 FORMAT(1X,33A1,12)
  4 CONTINUE
  READ(IN,106) (ROTUL(I,21),I=1,20),IAG(N)
106 FORMAT(20A1,12)
  WRITE(OUT,107) (ROTUL(I,21),I=1,20),IAG(N)
107 FORMAT(1X,20A1,12,' A"DS')
  GOTO 6
  5 CONTINUE
  NGE(N)=NGE(N-1)
  IAG(N)=IAG(N-1)
  6 CONTINUE
  READ(IN,108,END=10) ICOD(N),(ROTUL(I,N),I=1,77)
108 FORMAT(2X,11,77A1)
  WRITE(OUT,109) ICOD(N),(ROTUL(I,N),I=1,77)
109 FORMAT(' 1-',11,77A1)
  7 CONTINUE
  IF(IFMT1.NE.1) GOTO 8
  NL=NGE(N)
  READ(IN,110) (DATOS(I,N2M1),I=1,NL)
  WRITE(OUT,111) (DATOS(I,N2M1),I=1,NL)
  READ(IN,110) (DATOS(I,N2),I=1,NL)
  WRITE(OUT,111) (DATOS(I,N2),I=1,NL)
110 FORMAT(8F10.0)
  GOTO 9
  8 CONTINUE
  NL=NGE(N)
  READ(IN,FMT) (DATOS(I,N2M1),I=1,NL)
  WRITE(OUT,111) (DATOS(I,N2M1),I=1,NL)
  READ(IN,FMT) (DATOS(I,N2),I=1,NL)
  WRITE(OUT,111) (DATOS(I,N2),I=1,NL)
  9 GOTO 1
  111 FORMAT(1X,13F10.0)
C
C*****
C
C LLAMADA A LA SUBROUTINA DE TRANSFORMACION DE LOS DATOS INICIALES.
C
  10 CONTINUE
  N=N-1
  CALL TRANSF(N)

```

C  
C .....  
C  
STOP  
END

### 1.3.11. Algunos ejemplos

Las pirámides que se presentan a continuación explicitan gráficamente las prestaciones más importantes del programa (Fig. 1.27., 1.28., 1.29. y 1.30.).

### 1.4. DIBUJO AUTOMÁTICO DE DIAGRAMAS TRIANGULARES. EL PROGRAMA TRIAN.

El empleo del diagrama triangular en trabajos de investigación geográfica, si bien no alcanza las cotas de la pirámide de población, no deja de ser importante. Por este motivo y por la dificultad de su confección, sobre todo cuando se pretende una aceptable precisión, nos ha parecido interesante elaborar un programa que lo realice automáticamente.

El programa TRIAN, por el momento, no es nada más que esto: un programa de representación gráfica. No se contempla en él ningún otro procedimiento de clasificación de los casos que el que resulta de la mera inspección visual del dibujo. Este hecho limita su uso a los supuestos en los que los individuos se agrupan claramente en zonas muy contrastadas del diagrama. Para resolver el problema de la definición de tipologías cuando no se produce esta situación, se han propuesto varios métodos de disección de la superficie del diagrama triangular -por ejemplo, Bertin /12/-, pero éstos no han sido por el momento incorporados a este procedimiento automático.

La construcción de un diagrama triangular tiene sentido siempre que se pretenda clasificar un determinado número de individuos en función del diferente peso sobre ellos de tres características -normalmente se trata siempre de porcentajes- cuya suma es siempre constante. En Geografía se ha empleado cuantas veces se ha pretendido clasificar un conjunto de ámbitos espaciales, en función de diversas estructuras (demográfica, productiva, cultural, etc.) explicitadas esquemáticamente por la importancia relativa de tres componentes, que, en cada caso, se definen previamente.

#### 1.4.1. Los términos de la representación

Tres tareas fundamentales pueden distinguirse en la elaboración de un diagrama triangular: dibujo del triángulo equilátero, rotulación coherente de los ejes -lados- del diagrama y localización en el interior del diagrama de los puntos que representan la caracterización de los casos del problema.

La primera de ellas es claramente trivial y no merece ningún comentario.

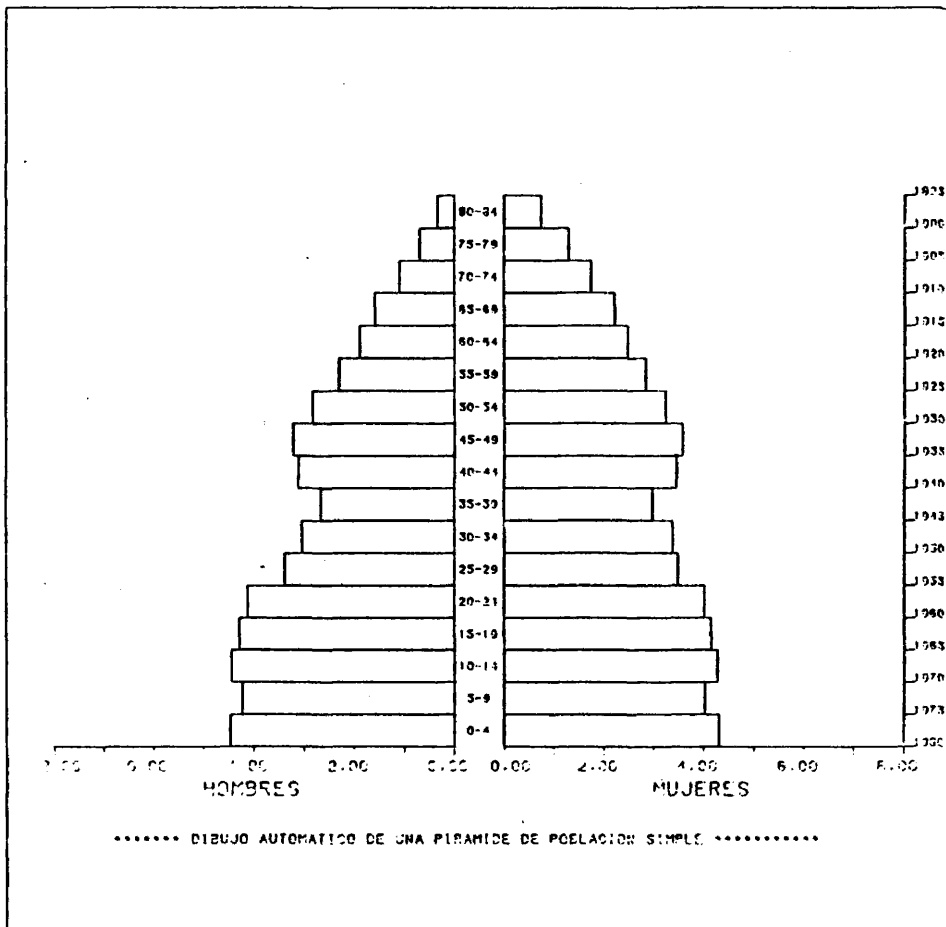


Fig.1.27

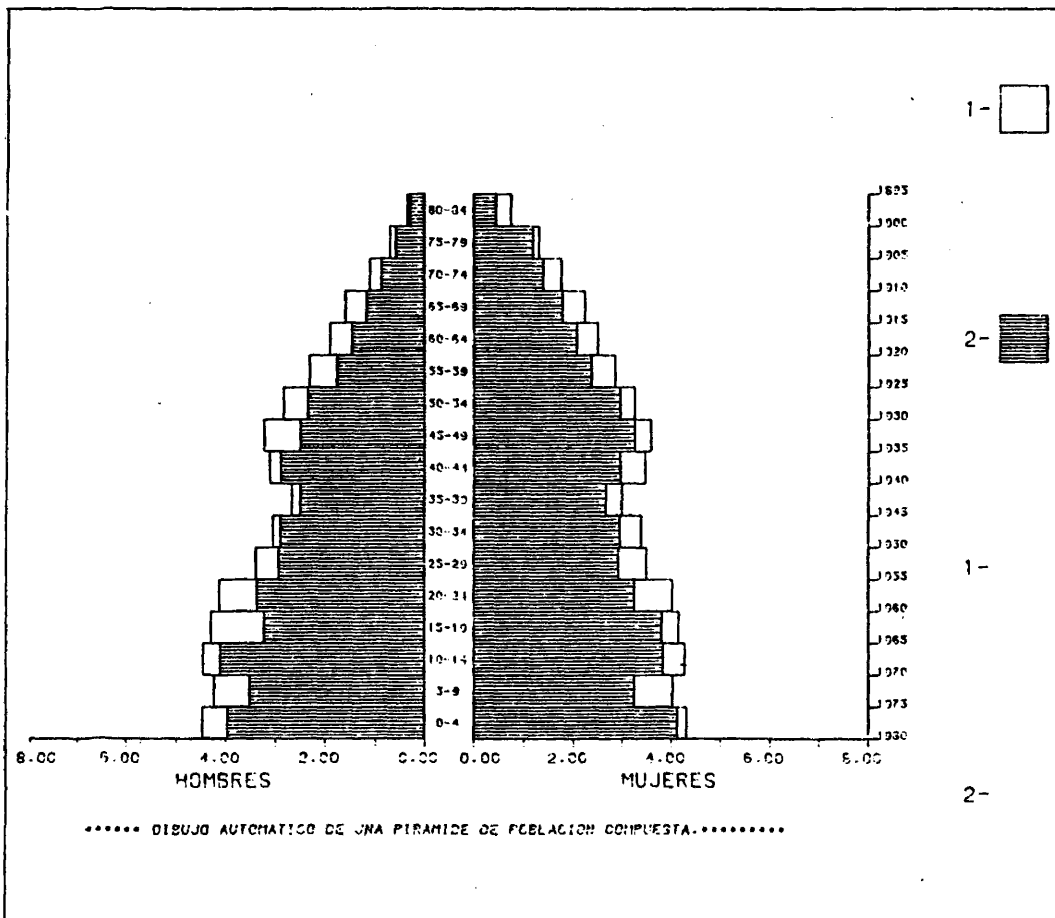


Fig1.28.



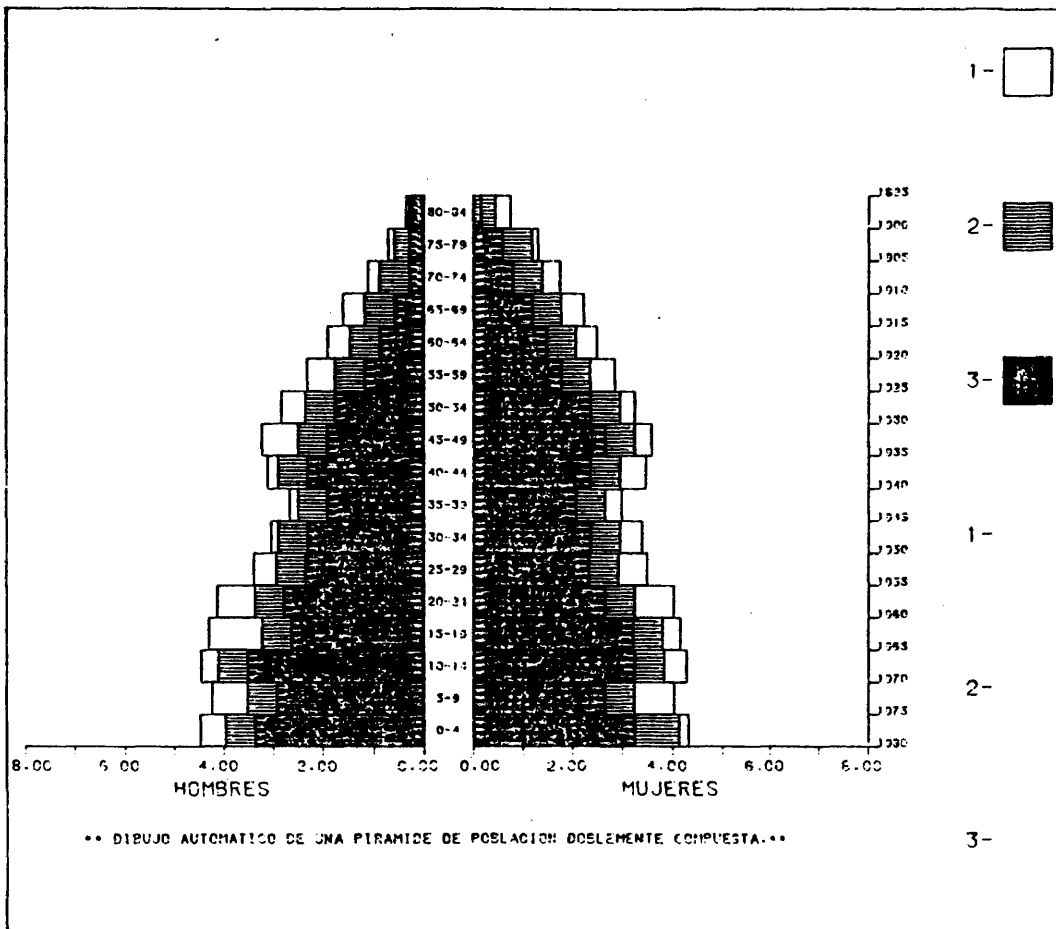


Fig.1.29.

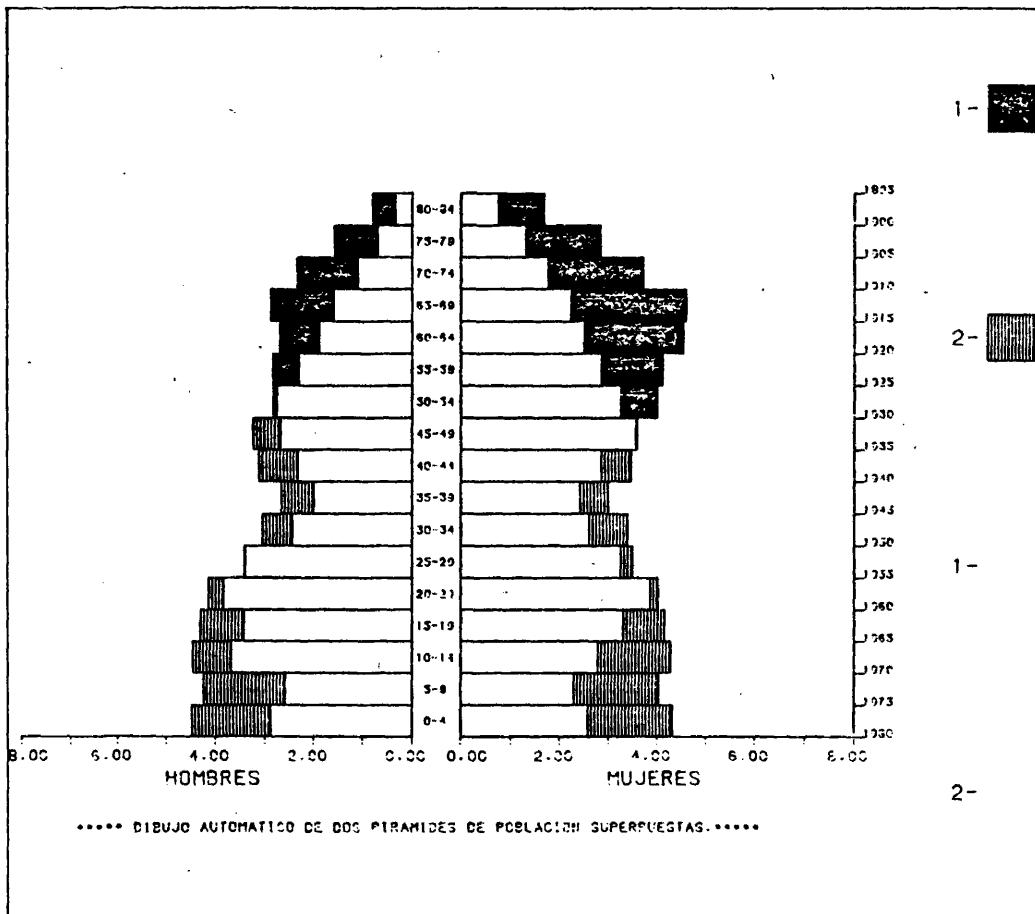


Fig.1.30.

La rotulación de los ejes del diagrama presenta un primer problema, en el sentido de que no existe unanimidad en la forma en que ésta ha de realizarse. Así, mientras en la mayoría de los casos se procede a una división significativa de los lados del triángulo equilátero y a la rotulación coherente de los valores que se corresponden con esas divisiones, en otros casos -Bertin/12/, por ejemplo- se rotulan los valores que corresponden a las bases de las tres alturas y a los tres vértices del triángulo. En el caso que nos ocupa, aun reconociendo que la propuesta de Bertin es la más correcta -ya que son las alturas del triángulo los auténticos ejes de la variación en el interior del diagrama-, se ha optado por la primera solución, por ser ésta la más difundida en la literatura geográfica.

Finalmente, para la localización de los casos en el interior del diagrama se ha recurrido a dibujar todas las divisiones significativas de la superficie de éste y a situar un signo convencional sobre los puntos que se corresponden con ellos.

Para enmarcar el dibujo y situar sobre el gráfico un rótulo significativo de 240 caracteres máximo se ha recurrido a dos subrutinas muy comunes -MARCO Y TITULO-, que también utilizan otros programas que presento en esta tesis.

#### 1.4.2. Limitaciones actuales del procedimiento.

Junto a las señaladas un poco más arriba, al hablar de diversas alternativas de clasificación, hay que destacar dos limitaciones de la versión actual del programa.

Me refiero, por una parte, al hecho de que requiere conocer, como datos de partida, los pesos de cada variable en cada caso en forma de porcentajes sobre el total del peso que afecta a cada caso. Esto supone que el usuario ha de hacer previamente esta transformación. De cualquier manera no se trata de una limitación grave, ya que se puede solucionar fácilmente, sin modificar el programa, diseñando una subrutina muy sencilla de transformación de los datos de entrada.

En segundo término es necesario, cuanto antes, flexibilizar la sección de rotulación de los ejes del diagrama, ya que por el momento el programa sitúa, en todos los casos, bajo el lado horizontal del triángulo el rótulo "JOVENES", en el margen lateral derecho del rótulo "ADULTOS", y en el izquierdo el rótulo "VIEJOS". Lógicamente este problema se resolverá pidiendo al usuario que introduzca los nombres de las variables que se corresponden con cada uno de los lados del triángulo, situándolos posteriormente en el lugar apropiado. Se trata de algo trivial, pero mientras no se haga, el programa sólo puede ser utilizado para representar grupos de población en función de su estructura en grandes grupos de edad.

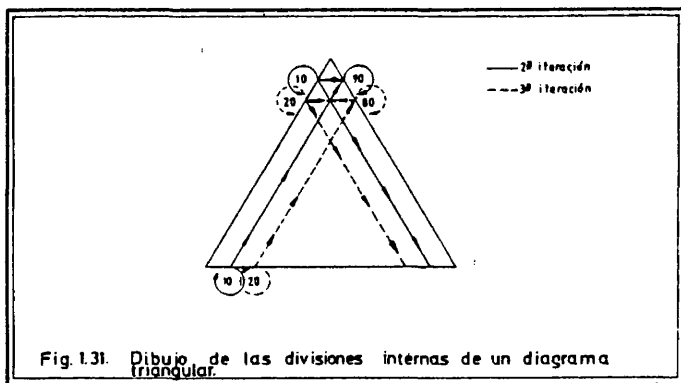
### 1.4.3.- Construcción automática del diagrama triangular.

El dibujo del diagrama triangular está confiado a la subrutina TRIANG, que parte de los datos a presentar y de una serie de optativos de dibujo que el usuario puede controlar para obtener la representación más adecuada a sus necesidades.

Vamos a considerar ahora las secciones fundamentales de la subrutina en su versión standard, ya que en el próximo apartado se describen las posibilidades de obtención de diagramas condicionados por las peticiones del usuario.

En primer lugar, se dibuja un triángulo equilátero proporcional a las dimensiones del cuadro que va a enmarcar el diagrama, que, en la versión standard, se supone va a contemplar todas las posibilidades de variación interna -esto significa que las rotulaciones de los lados han de cubrir el intervalo 0%, 100%-.

A continuación se procede a rotular adecuadamente los lados del triángulo y a dibujar coherentemente las divisiones internas del diagrama, para facilitar la lectura del gráfico. Se trata de un solo procedimiento iterativo que minimiza los desplazamientos de la pluma necesarios para la realización de esta tarea. En la figura 1.31. se describen gráficamente los movimientos de la pluma en las segunda y tercera iteraciones.



Se recurre en esta sección del dibujo a la utilización combinada de las subrutinas PLOT y NUMBER, del software del sistema CALCOMP, y a la subrutina INTERS, que calcula los puntos de corte de las divisiones internas con los lados del triángulo exterior.

Posteriormente se sitúan adecuadamente las leyendas de los tres lados del triángulo. En el momento presente se reproducen in defectiblemente los rótulos "JOVENES", "ADULTOS" y "VIEJOS", como ya hemos comentado líneas arriba.

Después de definir el vértice inferior izquierdo del triángulo como nuevo origen de coordenadas y de calcular las componentes ortogonales de los vectores unitarios, que definen las direcciones de variación de las dos primeras características -% de jóvenes y % de adultos, en este caso-, se procede a calcular las coordenadas de los puntos que se corresponden con cada caso del problema y se sitúa sobre ellos un signo centrado convencional -una mota circular de 1 mm de radio- recurriendo a la subrutina MOTA.

Finalmente, se reproduce sobre el diagrama el literal -de 240 caracteres máximo- que ha introducido el usuario y se enmarca el dibujo.

#### 1.4.4. Control de la legibilidad del gráfico resultante.

Posteriormente a la redacción de la versión standard de la subrutina TRIANG, se han introducido varios cambios en ella para permitir al usuario un mayor control sobre el dibujo resultante.

Básicamente, el usuario dispone de tres mecanismos de control para mejorar la legibilidad del gráfico, en caso de que la solución standard no sea satisfactoria en un momento dado.

Así, en primer lugar, si se desea, se puede obtener tan sólo la reproducción de la parte del diagrama en que se localizan los casos del problema.

En segundo término, el usuario puede decidir ampliar o reducir el tamaño de cualquier dibujo ya reproducido, mediante un operativo simple. Por ejemplo, si se desea obtener una reproducción a tamaño doble de un diagrama ya dibujado, el usuario deberá asignar a este parámetro el valor 2; si se pretende una reducción del dibujo a la mitad de su tamaño, el valor a introducir será 0.5, y así sucesivamente.

Finalmente, para poder identificar la relación que existe entre los puntos significados en el interior del diagrama y los casos del problema, el usuario tiene la posibilidad de reproducir un diagrama de idénticas dimensiones, en el que no se rotulan los ejes ni se dibujan las divisiones internas y en el que sobre los puntos que se corresponden con cada caso se sitúa, en vez de una mota circular, el número de orden del caso en la tabla estadística de referencia.

Aunque no está actualmente disponible, se trabaja en el momento presente en el desarrollo de nuevos procedimientos que permitan al usuario seleccionar fácilmente para cada caso el tipo de

signo convencional que lo va a representar en el interior del diagrama. Esta ampliación del procedimiento permitirá caracterizar cualitativamente a los casos del problema en función de alguna otra característica, y obtener una correlación visual de la distribución de dos características.

#### 1.4.5. Listado del programa

El listado del programa, que presentamos a continuación, comprende -en forma de comentarios- todas las instrucciones de uso y una descripción detallada de cómo han de organizarse los datos que aporta el usuario.

Las personas interesadas en la estructura del programa de dibujo, habrán de recurrir a la consulta del apéndice final de subrutinas, ya que el programa principal que sucede a estas líneas no es en sí mismo más que un programa de lectura de los datos de base.

```

*****
* PROGRAMA DE LECTURA DE LOS DATOS DE ESTRUCTURA POR GRANDES GRU-
* POS DE EDAD, EN PORCENTAJES, DE DIVERSAS POBLACIONES (99 COMO
* MAXIMO (1)), QUE SE VAN A REPRESENTAR GRAFICAMENTE MEDIANTE UN
* DIAGRAMA TRIANGULAR.
*
* *****
* (1) SI SE QUIERE USAR ESTE PROGRAMA PARA TRABAJAR CON MAS DE 99
* CASOS, HAY QUE REDIMENSIONAR LA MATRIZ DAT.
*****
*****
NOMBRE DEL PROGRAMA: TRIAN.
LENGUAJE: FORTRAN77
AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
DEPARTAMENTO DE GEOGRAFIA HUMANA.
UNIVERSIDAD COMPLUTENSE (MADRID).
JULIO 1982.
ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
MEMORIA: 148 KBYTES.
SUBROUTINAS NECESARIAS: TRIANG, INTERS, MODA, TITULO Y MARCO.
*****
*****
ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
PRIMERA TARJETA.
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
//FT05F001 DD *
SEGUNDA TARJETA.
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
AAA
A = NUMERO DE CASOS DEL PROBLEMA.
TERCERA TARJETA (POR EJEMPLO).
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
(3F10.0)
DESCRIPCION DEL FORMATO DE LOS DATOS (PUEDE SER CUALQUIERA).

```

C CUARTA TARJETA (O TARJETAS): DATOS DE ESTRUCTURA POR GRANDES GRUPOS  
 C DE EDAD DE LAS A POBLACIONES, GRABADOS  
 C CON EL EL FORMATO QUE SE ACABA DE  
 C DESCRIBIR.

C LA LECTURA ES CASO A CASO, Y EL ORDEN DE LAS VARIABLES ES :JOVENES,  
 C ADULTOS Y VIEJOS.

C SUPONIENDO QUE EL FORMATO ES EL DEL EJEMPLO Y QUE SE CONSIDERAN DOS  
 C POBLACIONES, LOS DATOS SE PRESENTAN DE LA SIGUIENTE MANERA:

COL.	1	10	20	30	40	50	70	80
1	10	20	30	40	50	70	80	
	30	20	50					
	45	25	30					

C QUINTA TARJETA.

COL.	1	10	20	30	40	50	70	80
1	10	20	30	40	50	70	80	
B								
C								

C SI B = 0, SE DIBUJA TODO EL DIAGRAMA.  
 C SI B = 1, SOLO SE DIBUJA LA PARTE DEL DIAGRAMA AFECTADA.  
 C SI C = 0, SE SITUAN NOTAS CIRCULARES EN EL INTERIOR DEL DIAGRAMA.  
 C SI C = 1, SE SITUAN NUMEROS CORRELATIVOS.  
 C SI D = 1, EL DIBUJO QUEDA ENMARCADO POR UN CUADRO DE 20\*20 CMS.  
 C SI D ES DISTINTO DE 1, LAS DIMENSIONES DEL DIBUJO SE MULTIPLICAN  
 C POR D.

C SEXTA TARJETA.

COL.	1	10	20	30	40	50	70	80
1	10	20	30	40	50	70	80	
EEE								

C E = NUMERO DE TARJETAS (MAXIMO 3) SOBRE LAS QUE ESTA ESCRITO EL  
 C TITULO DEL GRAFICO.

C SEPTIMA TARJETA (O TARJETAS).

COL.	1	10	20	30	40	50	70	80
1	10	20	30	40	50	70	80	
DIAGRAMA TRIANGULAR QUE REPRESENTA.								

C ULTIMA TARJETA.

COL.	1	10	20	30	40	50	70	80
1	10	20	30	40	50	70	80	
/*								

C \*\*\*\*\*  
 C \*\*\*\*\*  
 C \*\*\*\*\*

C DIMENSION DAT(3,99)  
 C CHARACTER FMT (80)  
 C INTEGER OUT

C DAT = MATRIZ DONDE SE ALMACENAN LOS DATOS DE ESTRUCTURA QUE SUMINIS-  
 C TRA EL USUARIO.  
 C FMT = VECTOR DE CARACTERES QUE SE UTILIZA PARA DESCRIBIR EL FORMATO  
 C DE LOS DATOS QUE APORTA EL USUARIO.

C \*\*\*\*\*  
 C \*\*\*\*\*



```

C  INICIALIZACION DE LAS VARIABLES DE LECTURA Y ESCRITURA.
C
    DATA IN,OUT/5,6/
    DATA INFDR/2/
C
C  LOS DATOS QUE DEFINE EL USUARIO CONSTITUYEN EL FICHERO FT05F001.
C
C*****
C  LECTURA DE DATOS
C
    READ(IN,100)N
100 FORMAT(5X,13)
    IF(INFDR.NE.1)
      *WRITE(OUT,100)N
    READ(IN,101)(FMT(I),I=1,80)
101 FORMAT(80A1)
    IF(INFDR.NE.1)
      *WRITE(OUT,102)(FMT(I),I=1,80)
102 FORMAT('D EL FORMATO DE LOS DATOS ES = ',80A1)
    READ(IN,FMT)((DAT(I,J),I=1,3),J=1,N)
    IF(INFDR.NE.1)
      *WRITE(OUT,103)(J,(DAT(I,J),I=1,3),J=1,N)
103 FORMAT(10X,'J',9X,'A',9X,'V'/(13,3F10.2))
    READ(IN,104) IAMPL, IOPT, FCT
    IF(INFDR.NE.1)
      *WRITE(OUT,105) IAMPL, IOPT, FCT
104 FORMAT(11,8X,11,F10.0)
105 FORMAT(5X,11,8X,11,F10.0)
C
C*****
C  LLAMADA A LA SUBROUTINA DE DIBUJO.
C
    CALL TRIANG(DAT,N,IAMPL,IOPT,FCT)
C
C*****
C
    STOP
    END

```

#### 1.4.6. Algunos ejemplos.

Los diagramas que se presentan a continuación, a partir de un conjunto de datos experimentales, son una plasmación de las diversas prestaciones del programa que acabamos de comentar (Figs. 1.32., 1.33. y 1.34.).

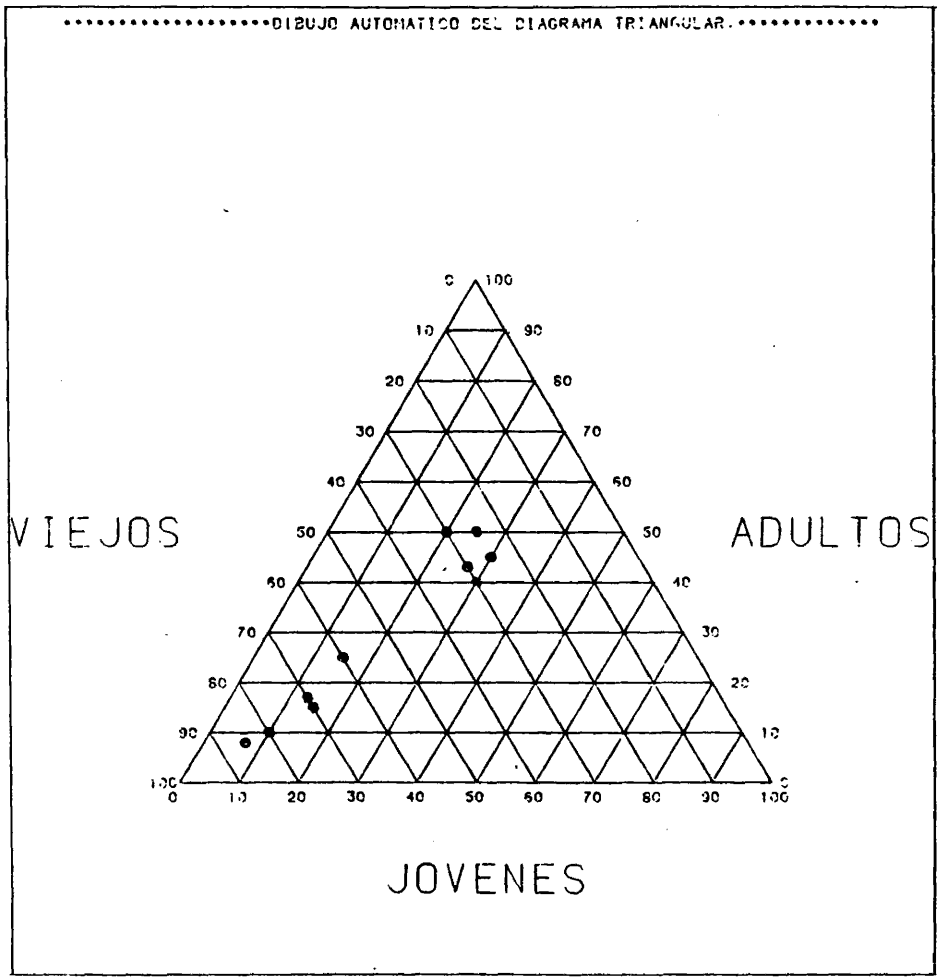


Fig.132.

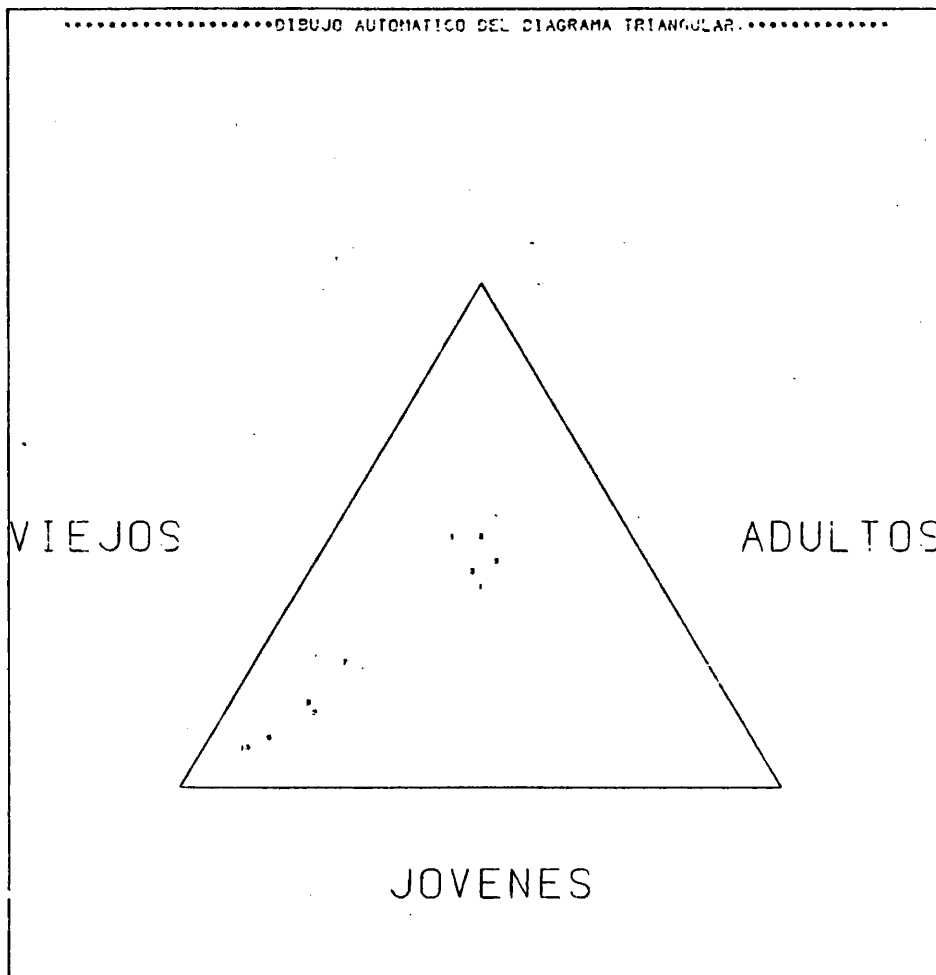


Fig.1.33.

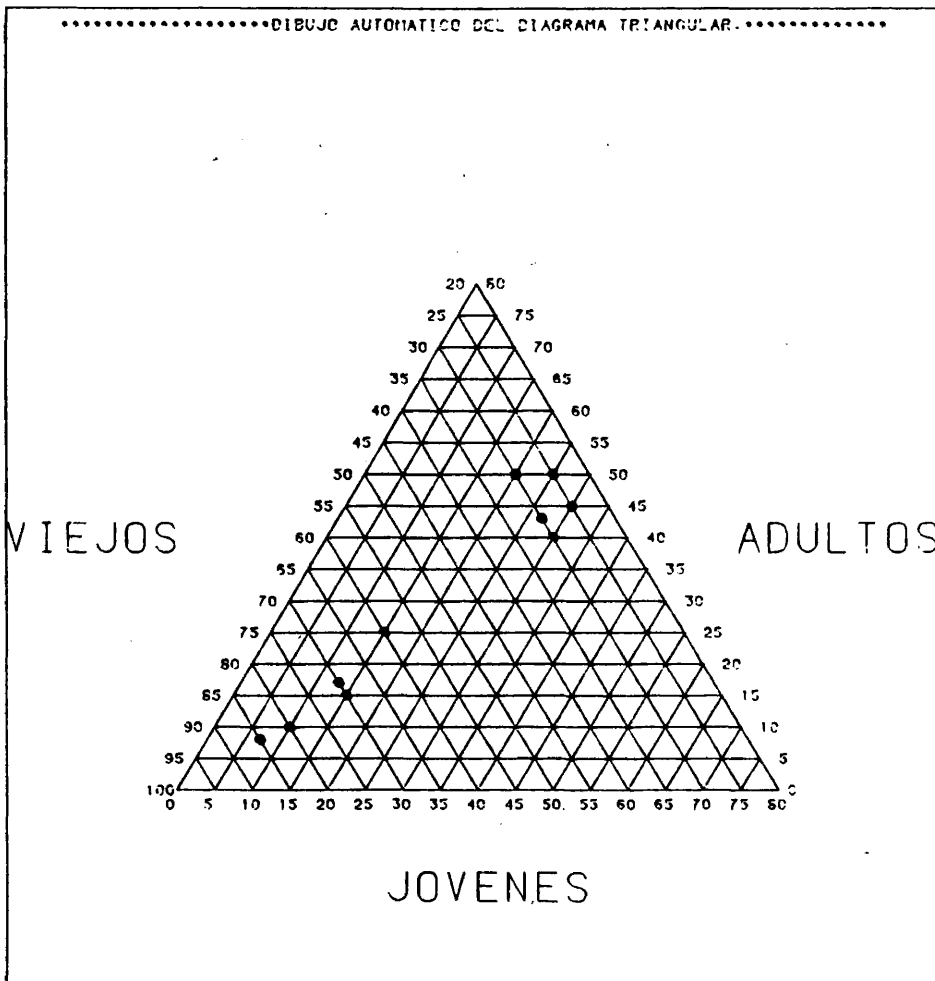


Fig.134.

NOTAS AL CAPITULO 1.

(1) En la Memoria de Licenciatura a que nos estamos refiriendo se ha resuelto la elaboración automática mediante ordenador -utilizando como periférico gráfico un plotter de pluma CALCOMP 563- de los siguientes diagramas:

- . diagrama circular trámado.
- . climograma de precipitaciones y temperaturas medias mensuales.
- . climo-diagrama de Confort.
- . diagrama lineal de evolución temporal comparada de dos variables.

(2) En el catálogo de programas disponibles en el GPE -Geography Program Exchange-, que es una guía bastante completa de aplicaciones informáticas a la investigación geográfica, aparecen dos procedimientos que también realizan automáticamente pirámides de población. El programa POPPYR, desarrollado en 1975 por el Departamento de Geografía del "Virginia Polytechnic Institute", reproduce pirámides de población, aunque el catálogo no indica qué tipo de periférico gráfico utiliza. El programa PYRAMID del Departamento de Geografía de la "Pensylvania State University", redactado en 1972, realiza pirámides de población en la impresora de líneas de un ordenador. No existe entre estos programas y el que presento en este capítulo otra coincidencia que el hecho de causar el mismo tipo de representación. El programa PIRAMID realiza, además, todo un conjunto de tareas que estos dos programas omiten: realización de pirámides compuestas, superposición de pirámides, representación de la escala de años de nacimiento de las cohortes de la pirámide, etc.

Por lo que respecta al programa de confección automática de diagramas triangulares, no he encontrado ningún paralelismo en la literatura sobre el tema.

(3) Para el caso concreto de los ejemplos de cartogramas automáticos que presento en esta tesis, he contado con la ayuda considerable de Joaquín Bosque Sendra y Beatriz Jiménez Blasco, que ya disponían de digitizaciones de contornos geográficamente significativos que habían procesado con otros programas de cartografía automática ya disponibles en Madrid -SYMAP, fundamentalmente- antes del comienzo de esta tesis.

(4) Es por este motivo por el que he aceptado gustosamente la invitación de participar en un proyecto de investigación interdisciplinar, que se ha formulado por iniciativa de la Dirección General del IGN y que, bajo el título "Sistema de trata

miento automático de la información espacial para la planificación física y la ordenación del territorio", pretende la definición de una base de datos espacialmente referenciados y su explotación por programas de representación gráfica y de análisis estadístico-espacial, con vistas a obtener criterios ciertos de planificación física y socioeconómica.

- (5) Mi interés por los gráficos por computadora tuvo origen, hace ya tres años, en el marco de una investigación que, aparentemente, no tenía ninguna relación con este tema.

Joaquín Bosque Sendra y yo estábamos adaptando y ampliando un programa de ordenador de John P. Cole para establecer proyecciones de población a medio plazo, en base a la estructura por edad y sexo y a las tasas específicas de mortalidad y fecundidad de un colectivo humano.

Conforme nos íbamos adentrando en el tema, surgían nuevas posibilidades de refinamiento y mejora del modelo de proyección y de su versión informática. En este sentido, comprendimos en seguida que era necesario introducir dentro de él la opción de considerar los flujos migratorios, si se disponía de datos fiables acerca de ellos.

En su vertiente demográfica, este programa ya ha sido utilizado en diversos objetivos de investigación (entre otros, Requés /133/), y en la actualidad se encuentra disponible en el Centro de Cálculo de la Universidad Complutense. De hecho, recurren a él con mucha frecuencia los estudiantes de últimos cursos de Licenciatura, para establecer proyecciones a medio plazo de las poblaciones provinciales españolas.

Teniendo en cuenta que este programa parte de la estructura por edad de una determinada población, y la proyecta hacia el futuro -admitiendo diversas hipótesis de evolución futura de las tasas específicas de fecundidad y mortalidad, en saltos discretos, recalculando todos los parámetros que describen una determinada población- cada cierto número de años, en función de una orden del usuario, que puede decidir sobre la magnitud de dichos saltos, nos pareció de gran interés obtener una imagen automática de la estructura por edad y sexo de la población, en el momento inicial, en el final, y en todos los pasos intermedios.

Así me dediqué a diseñar una subrutina gráfica (PYRAM), que era llamada desde el programa principal en cada etapa de la proyección, para obtener una imagen plástica de la estructura por edad y sexo de la población.

Esta subrutina utiliza la impresora de líneas para obtener el gráfico deseado. No me voy a detener en su explicación por menorizada, por el hecho de que ya existe una versión mucho más precisa de este objetivo de representación gráfica. Tan

sólo me he referido a ella para dejar constancia de que en la génesis de la presente tesis ha constituido el auténtico punto de partida.

El programa de proyección de población a que me estoy refiriendo también utiliza otra subrutina gráfica (GRAFIC) para obtener un diagrama bivariado, que representa la evolución temporal futura de la población.



CAPITULO 2. CARTOGRAFIA TEMATICA EN DOS DIMENSIONES: MAPAS DE  
COROPLETAS.

	<u>Págs.</u>
2.1. INTRODUCCION .....	95
2.2. DESCRIPCION DIGITAL DE AREAS EN TERMINOS VECTORIALES	96
2.2.1. Alternativas de digitización de áreas.....	97
2.2.2. Reconstrucción automática de contornos a partir de líneas de frontera interáreas .....	100
2.2.3. Fusión de áreas con contigüidad espacial ....	106
2.2.4. Reconstrucción de contornos y jerarquía espacial. Dos alternativas. ....	111
2.2.4.1. El programa POLIGS3. ....	112
2.2.4.2. El programa POLIGS4. ....	117
2.2.5. Verificación y corrección de trabajos de digitización de áreas .....	126
2.3. RAYADO DE POLIGONOS .....	130
2.3.1. Cálculo de los puntos de intersección de una línea recta con una poligonal cerrada. Solución de los problemas de límite y colinealidad	130
2.3.2. Intersección de una línea recta con diversas poligonales. Solución al problema de rayado de polígonos concéntricos .....	132
2.3.3. Definición de tramas simples .....	134
2.3.4. Definición de tramas compuestas .....	137
2.4. ELABORACION AUTOMATICA DE MAPAS DE COROPLETAS.....	138
2.4.1. Definición precisa de las tareas a acometer..	138
2.4.2. Problemas de clasificación que están relacionados con la construcción de un mapa de coropletas .....	140
2.4.3. El programa COROMAP. ....	142
2.4.3.1. Articulación de las rutinas de clasificación y de dibujo .....	142
2.4.3.2. Información gráfica adicional que proporciona .....	144
2.4.3.3. Listado del programa y requisitos de uso del mismo .....	144
2.4.3.4. Algunos ejemplos .....	152

## 2.1. INTRODUCCION.

Para llegar a la definición precisa y operativa de un procedimiento automático de construcción de mapas de coropletas, es necesario tener en cuenta un conjunto muy amplio de cuestiones que están involucradas en dicha definición. Muchas de ellas tienen entidad suficiente para constituir un apartado dentro de este capítulo, sobre todo las relativas a la definición de áreas en términos digitales y a la solución del problema de tramado de polígonos.

La estructura de este segundo capítulo tiene en cuenta lo que acabo de señalar, abordando sucesivamente todos los problemas que ha sido necesario resolver y subrayando aquéllos que tienen una mayor relevancia por sí mismos, o por su conexión con otros ámbitos de la cartografía digital.

Si bien el planteamiento y la resolución de los problemas que se abordan en este capítulo son básicamente personales, es imprescindible señalar en este momento los paralelismos que existen entre los resultados que presento ahora y los de toda una serie de programas desarrollados, fundamentalmente en EE.UU., desde mediados de los años 60 hasta la actualidad.

Los primeros programas de establecimiento automático de mapas de coropletas, utilizan como periférico gráfico la impresora de líneas. Entre ellos destaca, por su difusión, el programa SYMAP, cuya primera versión aparece en 1966, desarrollado por el "Harvard Laboratory for Computer Graphics and Spatial Analysis". Otros programas disponibles que realizan esta misma tarea son: CMAP, desarrollado por el Departamento de Geografía de la Universidad de Idaho en 1971; CPLETH, desarrollado por el Departamento de Geografía de la Universidad de Michigan en 1971; MINIMAP, desarrollado por el Departamento de Geografía y Planificación Territorial del "Towson State College", en 1973; COBMAP, desarrollado por el Departamento de Geografía de la Universidad de Tennessee, en 1976 y XMAPP, desarrollado por el Departamento de Geografía de la Universidad de Illinois.

Con la aparición en escena de otros periféricos gráficos utilizables -plotters, pantallas de rayos catódicos, transformadores cinta/película, etc.-, comienza el desarrollo de una nueva generación de programas de elaboración automática de mapas de coropletas de mucha mayor precisión y calidad. De la misma época, aunque haya sido mucho más difundido el segundo de ellos, son el programa INTRMAP, desarrollado en 1975 por el Departamento de Geografía de la "Virginia Commonwealth University", de elaboración interactiva de mapas de coropletas en una pantalla de rayos catódicos y el

programa CALFORM, desarrollado en 1976 por el "Harvard Laboratory for Computer Graphics and Spatial Analysis", que puede utilizar indistintamente un plotter de pluma, un plotter electrostático o una pantalla de rayos catódicos.

Más recientemente, se está asistiendo al desarrollo de una última generación de programas de diseño de mapas de coropletas incluidos en sistemas integrados de manipulación de información geográfica. En esta línea se encuentran los procedimientos de obtención de mapas de coropletas del sistema ODYSSEY -del "Harvard Laboratory for Computer Graphics and Spatial Analysis" y los incluidos en el sistema GIMMS -del Departamento de Geografía de la Universidad de Edimburgo-.

Por lo que respecta a las referencias bibliográficas más interesantes para el planteamiento y resolución de la obtención automática de mapas de coropletas, destacan, por orden cronológico, los trabajos de Monmonier /116/, Wilson /180/, Monmonier /118/, Howart /79/, Jeffery, O'Hare y Board /82/ y McDougall /97/, cap.5.

## 2.2. DESCRIPCION DIGITAL DE AREAS EN TERMINOS VECTORIALES.

Brevemente, el problema del establecimiento de un mapa de coropletas se reduce al sombreado diferencial de un conjunto de polígonos, coherente con el peso de una determinada característica en el interior de cada uno de ellos, siempre que dichos polígonos constituyan una representación, más o menos aproximada, de unidades geográficamente significativas.

Cuando este trabajo se lleva a cabo manualmente, las capacidades humanas pueden obtener un reconocimiento visual de las zonas a tramar y reproducir dentro de ellas el sombreado predeterminado, que simboliza la magnitud del peso que las afecta.

Si se trata, en cambio, de lograr que un determinado periférico gráfico gobernado por un ordenador digital realice esta tarea, es inevitable obtener una descripción en forma numérica de cada área a tramar.

La conversión de una imagen en su descripción numérica se conoce con el nombre de digitización -o digitalización-, como ya hemos comentado páginas arriba en esta tesis.

Para el caso concreto de un contorno cerrado, su descripción numérica vectorial es el conjunto de pares de coordenadas cartesianas que definen los puntos de una poligonal cerrada que aproxima aceptablemente su forma.

En este apartado vamos a centrarnos en diversos aspectos interesantes relacionados con la digitización de áreas, dejando pa-

ra los siguientes de este mismo capítulo los problemas de tramado de polígonos y los de conjunción de ambos -digitización y tramado- en la definición de un mapa de coropletas.

#### 2.2.1. Alternativas de digitización de áreas.

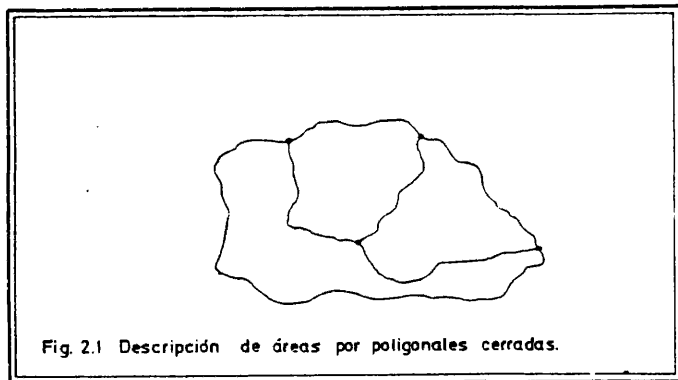
Sin pretender agotar ni mucho menos, este tema -pues es todo un tópicó en el campo de la cartografía digital-, vamos a considerar dos vías posibles de conversión en términos digitales de la imagen de un mosaico de unidades geográficamente significativas:

En ambos casos se requiere superponer sobre la imagen a digitalizar una retícula ortogonal que nos permita fácilmente evaluar las coordenadas de los puntos que definen las poligonales que aproximan los contornos, o utilizar un digitizador que grabe automáticamente dichas coordenadas al situar el cursor sobre cada uno de los puntos.

La divergencia entre ambos procedimientos se produce a la hora de decidir cómo y qué información espacial va a ser considerada

Lo más inmediato es definir cada contorno por el número de puntos que lo aproximan y por la lista consecutiva de todas las coordenadas de dichos puntos.

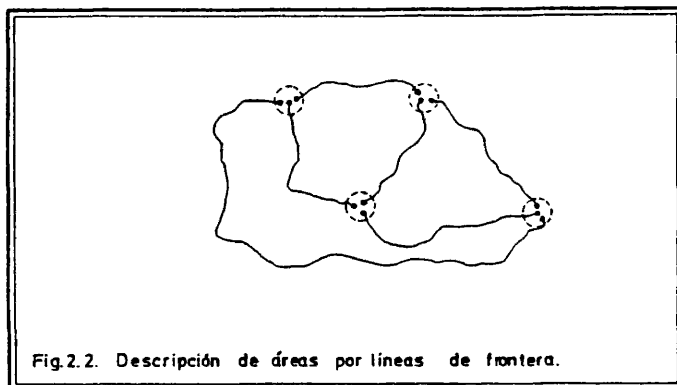
En el supuesto de una situación como la que representa la figura 2.1., la descripción digital de esa imagen estaría formada por tres conjuntos de datos, cada uno de los cuales constaría de un entero, indicando el número de puntos en cuestión, y de una sucesión de pares de coordenadas.



Ahora bien, si prestamos atención a la figura propuesta, resulta evidente que un procedimiento de este estilo resulta claramente antieconómico en cuanto a espacio de almacenamiento, ya que gran parte de la información digitizada, la correspondiente a los puntos que aproximan las tres fronteras, está duplicada -y triplicada la correspondiente a los nodos-.

Por este tipo de razones se han planteado nuevas estrategias de digitización de áreas, más económicas en cuanto al volumen de información a grabar. Una de ellas es la que vamos a considerar a continuación.

Si en lugar de grabar las coordenadas de los puntos de las tres poligonales cerradas, fragmentamos el dibujo en sus componentes más elementales -segmentos de línea entre dos nodos- y describimos cada uno de ellos por un número de puntos y las coordenadas de éstos, no se habrá producido ninguna duplicación de información, aunque sea inevitable la triplicación de las coordenadas de todos los nodos, de los que emergen, en este caso, tres segmentos (Fig. 2.2.).



Tenemos ahora seis conjuntos de datos -antes eran tres-, pero el volumen de información es claramente inferior. Si se trata de reproducir la figura en cuestión, el resultado es el mismo si se hace recorrer a una pluma todos los puntos grabados en el primer caso que si se efectúa la misma operación con los grabados en el segundo.

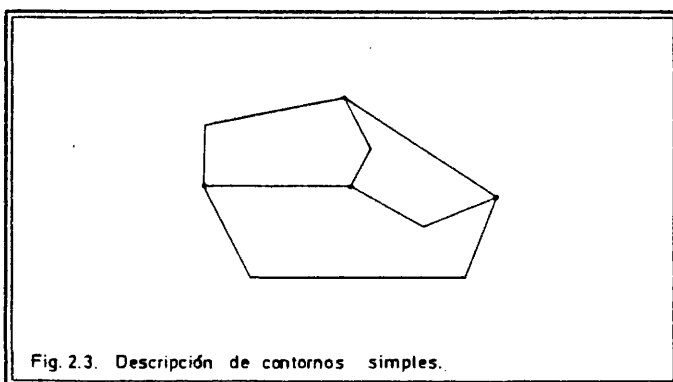
En cambio, si lo que pretendemos es realizar alguna operación con las zonas que se definen -como puede ser el tramado significativo de las mismas-, la situación cambia radicalmente ya que

la grabación realizada en el segundo caso -en su estado actual- no realiza una descripción digital satisfactoria de las áreas que se establecen. No obstante el problema puede ser resuelto de forma automática, recurriendo a algún tipo de procedimiento de reconstrucción de polígonos a partir de líneas, como se va a demostrar en las páginas siguientes.

Antes de finalizar estas consideraciones vamos a detenernos por un momento en la discusión de qué procedimiento es el más adecuado, teniendo en cuenta la configuración de la imagen a digitalizar.

En el caso de la imagen precedente las ventajas del segundo procedimiento son manifiestas. La razón de ello es que la configuración de los segmentos entre nodos es tal que requiere un elevado número de puntos para su descripción satisfactoria. Así las cosas, es preferible evitar la duplicación de varias listas de coordenadas de apreciable extensión.

Si prestamos atención, en cambio, a una situación como la que representa la figura 2.3., la escasa sinuosidad de los segmentos entre nodos -lo que supone que no necesitan ser descritos por un elevado número de puntos- puede llegar a justificar la descripción digital por poligonales cerradas, como alternativa más económica. Ya que en este caso se evitan los trabajos de reconstrucción de polígonos a partir de líneas de frontera, compensando así la redundancia -que tampoco es excesiva, en este supuesto- en la información almacenada.



Este caso, que a primera vista puede parecer irreal en su puesto de cartografía, puede producirse -de hecho hemos tenido

ocasión de comprobarlo- cuando se digitizan cuarteados administrativos que adoptan formas muy simplificadas.

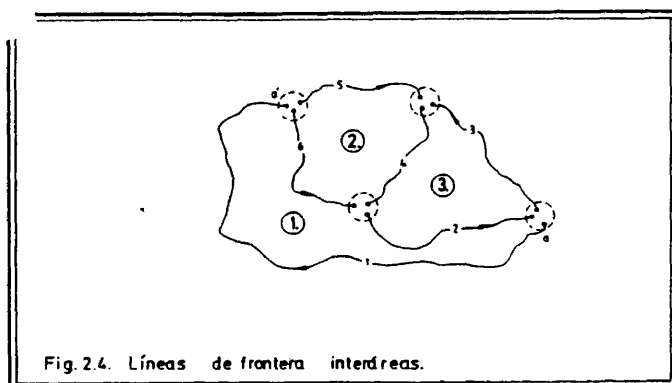
### 2.2.2. Reconstrucción automática de contornos a partir de líneas de frontera interáreas.

Aunque existen muy diversos procedimientos para lograr este objetivo, vamos a considerar sólomente el que hemos formulado en términos informáticos: el programa POLIGS1.

Los presupuestos del procedimiento que vamos a describir no difieren significativamente de los de otros algoritmos de reconstrucción de polígonos. No será difícil, por tanto, comprender el funcionamiento de otros procedimientos de esta índole, si se asimilan los requisitos del modelo que presentamos.

En primer lugar es necesario comentar que este procedimiento requiere para su adecuado funcionamiento un plus de información asociado a la descripción digital de las líneas de que se parte.

Si consideramos de nuevo la imagen a que ya hemos hecho referencia y asignamos a cada área a reconstruir un código numérico, nos encontramos con la siguiente situación (Fig. 2.4.).



Las flechas que hemos dibujado explicitan el sentido en que han sido digitizados los segmentos. Por ejemplo, en el caso del segmento  $aa'$ , la lista de pares de coordenadas de los puntos que lo aproximan se inicia con las correspondientes al punto  $a$  y finaliza con las correspondientes al punto  $a'$ .

El programa POLIGS1 necesita conocer, para cada línea, junto con el número de puntos que la constituyen y las coordenadas de las mismas, el código del área que deja a la derecha y el código del área que deja a la izquierda, de acuerdo con el sentido en el que haya sido digitizada.

El programa principal, que articula tres subrutinas: POLIG, INVERT y DEPURA, no es más que un programa de lectura de las líneas de que se parte, y de escritura en algún soporte informático de las poligonales cerradas que se reconstruyen. La subrutina POLIG es la que realiza propiamente la reconstrucción de cada área, en función de sus líneas de frontera, recurriendo, en su momento, a las subrutinas INVERT y DEPURA para resolver determinadas funciones.

En la figura 2.4. vamos a considerar esquemáticamente cómo funciona la subrutina POLIG en la reconstrucción del área 3., como ejemplo.

En primer lugar la subrutina investiga los códigos correspondientes a cada línea y sólo considera aquellas líneas en cuyo código compuesto aparece el identificador del área a reconstruir. En este caso concreto, las líneas que consideraría serían las 2, 3 y 4.

Conocidas las líneas que van a constituir la poligonal cerrada es necesario proceder a su adecuada concatenación. Para ello es necesario en primer lugar que todas las líneas estén descritas de tal forma que dejen el área en cuestión al mismo lado -en el procedimiento que hemos diseñado se ha adoptado la convención de que todas las líneas dejen el área a su derecha-. Para lograrlo se investiga cuál de los dos subcódigos es el que coincide con el identificador del área a reconstruir. Si es el código que significa "área a la derecha", la línea permanece inalterada -caso de la línea 4 en el ejemplo-. Si se produce la situación contraria -caso de las líneas 2 y 3- es necesario redescibir la línea en sentido inverso -esto lo realiza la subrutina INVERT- y cambiar después, el orden de los subcódigos.

Una vez estructuradas las descripciones de las líneas tal como acabamos de ver, se procede a su concatenación, considerando una de ellas, la primera en el orden general de las líneas -en este caso sería la 2-, y situando a continuación la línea cuyo origen es suficientemente próximo al final de aquélla -en este caso la 4-, repitiendo esta operación tantas veces como sea necesario -en el ejemplo una vez más-.

Finalmente, antes de devolver al programa principal la lista en la que se describe el polígono reconstruido, la subrutina POLIG llama a la subrutina DEPURA para eliminar de la descripción todos los puntos repetidos -previsiblemente los extremos de las líneas-.



Por sucesivas llamadas a la subrutina POLIG, el programa POLIGS1 reconstruye todas las áreas en cuestión. En el caso del ejemplo se llamaría a la subrutina POLIG tres veces.

Presentamos a continuación el listado del programa POLIGS1 para un estudio más detallado, si se desea. En el apéndice final de subrutinas aparecen los textos de POLIG, INVERT y DEPURA.

```

C
C *****
C * PROGRAMA DE EDICION DE FICHEROS DE AREAS EN LA UNIDAD 6, A PAR-
C * TIR DE LOS FICHEROS DE LAS LINEAS QUE LAS DEFINEN.
C * LOS FICHEROS DE LINEAS HAN DE TENER LA SIGUIENTE ORGANIZACION:
C * CODIGO DE AREA A LA DERECHA, CODIGO DE AREA A LA IZQUIERDA, NUME-
C * RO DE PUNTOS, X E Y DE CADA PUNTO.
C * *****
C * EL PROGRAMA CONSIDERA UN MAXIMO DE 100 LINEAS, DEFINIDAS POR UN
C * MAXIMO DE 100 PUNTOS Y UN AREA DEFINIDA POR UN MAXIMO DE 500 PUN-
C * TOS.
C * SI SE REBASAN ESTOS LIMITES HAY QUE REDIMENSIONAR COORD, POLY,
C * LAR E ICOD.
C * *****
C
C NOMBRE DEL PROGRAMA: POLIGSI.
C
C LENGUAJE: FORTRAN77.
C
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C MEMORIA: 168 KBYTES.
C
C SUBROUTINAS NECESARIAS: POLIG, INVERT Y DEPURA.
C
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN DOS FICHEROS.
C
C FICHERO FT05F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C AAB
C
C A = NUMERO DE AREAS A RECONSTRUIR.
C B = NUMERO DE LINEAS DE LAS QUE SE PARTE.
C
C TERCERA TARJETA (COMO EJEMPLD).
C 1 10 20 30 40 50 70 80

```

```

COL.1.....|.....|.....|.....|.....|.....|.....|
C (315,13F5,0(16F5,0))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS
C CODIGOS DE CADA LINEA, EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS.
C
C CUARTA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL.1.....|.....|.....|.....|.....|.....|.....|
C (214/(8F10,3))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A ESCRIBIR EN LA UNIDAD 6
C EL CODIGO DE CADA AREA EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS QUE LA DESCRIBEN.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL.1.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT08F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL.1.....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C POR CADA LINEA.
C
C SEGUNDA TARJETA Y SIGUIENTES: CODIGO DE AREA A LA DERECHA, CODIGO
C DE AREA A LA IZQUIERDA, NUMERO DE PUNTOS Y COORDENADAS X E Y DE LOS
C PUNTOS, SEGUN EL FORMATO DESCRITO.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL.1.....|.....|.....|.....|.....|.....|.....|
C /*
C
C *****
C *****
C *****
C
C COMMON/A/COORD(200,100),POLY(500,2)
C COMMON/B/LAR(100),ICOD(2,100),NAR,NLIN--
C CHARACTER FMT(80)
C READ (5,100) NAR,NLIN
C 100 FORMAT(2I2)
C READ(5,200) (FMT(I),I=1,80)
C 200 FORMAT(80A1)
C DO 1 I=1,NLIN
C
C SE LEEN LAS COORDENADAS DE LOS PUNTOS DE LAS LINEAS Y LOS CODIGOS
C ASOCIADOS A ELLAS.
C
C READ(8,FMT) ICOD(1,I),ICOD(2,I),LAR(I),(COORD(J,I),J=1,LAR(I))
C 101 FORMAT(3I2,10F4,0)
C 1 CONTINUE

```

```
READ(5,200) (FMT(I),I=1,80)
DO 2 I=1,NAR
CALL POLIG(I,L1)
C
C SE ESCRIBE EL CODIGO DE UN AREA, EL NUMERO DE PUNTOS QUE LA DESCRIBEN Y LAS COORDENADAS DE LOS PUNTOS.
C
C WRITE(6,FMT)(POLY(J,1),POLY(J,2),J=1,L1)
C 2 CONTINUE
C
C .....
C
C STOP
C END
```

### 2.2.3. Fusión de áreas con contigüidad espacial.

Como desarrollo lógico de los procedimientos elaborados para la reconstrucción de polígonos a partir de líneas de frontera, nos pareció interesante ofrecer la posibilidad de reconstruir polígonos procedentes de fusiones de las unidades espaciales mínimas que constituyen la base de referencia para la digitización.

Dos situaciones posibles, en función de la existencia o no de una jerarquía espacial previamente definida, han sido consideradas en la resolución de este problema.

Vamos a describir a continuación el procedimiento elaborado para el caso en que no exista una jerarquización previa de las unidades espaciales de referencia, dejando para el siguiente apartado la discusión de los dos procedimientos que han sido desarrollados para la reconstrucción de polígonos, a partir de líneas, a diversos niveles de la jerarquía espacial.

El programa POLIGS2, que articula las subrutinas POLIG2, INVERT y DEPURA, realiza la reconstrucción de un polígono cerrado a partir de las líneas de frontera de dos unidades espaciales contiguas.

Para su funcionamiento requiere una información de base muy similar a la que necesita el programa POLIGS1, que acabamos de describir. Las líneas digitizadas deben incluir una información adicional -código de área a la izquierda, código de área a la derecha- y, además, en este caso, el programa ha de conocer tantos pares de códigos -los de cada pareja de áreas a fundir- como fusiones ha de realizar para reconstruir los polígonos cerrados que se solicitan.

Este procedimiento puede ser de gran utilidad en la reconstrucción de áreas compuestas que son el resultado, no de una jerarquía previamente establecida, sino de algún tipo de proceso de contagio espacial -difusión, relación funcional, pertenencia a un mismo intervalo de clasificación por características estructurales, etc.-.

En el caso del programa que estamos considerando, la subrutina POLIG2 es la que realmente efectúa la reconstrucción del polígono deseado. Se trata de una versión corregida de la subrutina POLIG, a la que acabamos de referirnos.

Brevemente, su funcionamiento es tal como sigue. Supongamos que se trata de reconstruir el polígono cerrado resultante de la fusión de las áreas 2. y 3. de la figura 2.4. En el programa principal se habrán almacenado en memoria las descripciones digitales de todas las líneas de frontera y sus códigos asociados, y se habrá leído un par de códigos -2, 3- que identifica las dos áreas de cuya fusión resulta el polígono que se desea reconstruir.

La subrutina POLIG2 investiga qué líneas de frontera ha de tener en cuenta para reconstruir el polígono 2-3. Esto lo resuelve seleccionando todas las líneas que cumplen el requisito de que uno de sus dos códigos coincide bien con el 2, bien con el 3 y de sechando la línea cuyos códigos se corresponden con las dos áreas a fundir -ya que es ésta la línea de frontera que hay que suprimir-. El resultado de esta operación es la consideración de, en el ejemplo, las líneas 2, 3, 5 y 6. El procedimiento de concatenación de éstas es el mismo que el que aparece en la subrutina POLIG, ya descrita.

A continuación presentamos el listado del programa POLIGS2, con la documentación necesaria para su utilización. El texto de la subrutina POLIG2 se encuentra en el apéndice final.

```

C
C *****
C * PROGRAMA DE EDICION DE FICHEROS DE AREAS EN LA UNIDAD 6, COMO *
C * RESULTADO DE FUSIONES DE PARES DE AREAS, A PARTIR DE LOS FICHEROS *
C * DE LAS LINEAS QUE LAS DEFINEN. *
C * LOS FICHEROS DE LINEAS HAN DE TENER LA SIGUIENTE ORGANIZACION: *
C * CODIGO DE AREA A LA DERECHA, CODIGO DE AREA A LA IZQUIERDA, NUME- *
C * RO DE PUNTOS, X E Y DE CADA PUNTO. *
C * *****
C * EL PROGRAMA CONSIDERA UN MAXIMO DE 100 LINEAS, DEFINIDAS POR UN *
C * MAXIMO DE 100 PUNTOS Y UN AREA DEFINIDA POR UN MAXIMO DE 500 PUN- *
C * TOS. *
C * SI SE REBASAN ESTOS LIMITES HAY QUE REDIMENSIONAR COORD, POLY, *
C * LAR E ICOD. *
C * *****
C
C NOMBRE DEL PROGRAMA: POLIGS2.
C
C LENGUAJE: FORTRAN77.
C
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C MEMORIA: 168 KBYTES.
C
C SUBRUTINAS NECESARIAS: POLIG2, INVERT Y DEPURA.
C
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN DOS FICHEROS.
C
C FICHERO FT05F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C AA
C
C A = NUMERO DE LINEAS DE LAS QUE SE PARTE.
C
C TERCERA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80

```

```

COL.....|.....|.....|.....|.....|.....|.....|.....|
C (315,13F5.0(16F5.0))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS
C CODIGOS DE CADA LINEA, EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS.
C
C CUARTA TARJETA (COMO EJEMPLO).
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C (214/(8F10.3))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A ESCRIBIR EN LA UNIDAD 6
C EL CODIGO DE CADA AREA EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS QUE LA DESCRIBEN.
C
C QUINTA TARJETA.
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C BB
C
C B = NUMERO DE FUSIONES A REALIZAR.
C
C SEXTA TARJETA Y SIGUIENTES (TANTAS COMO FUSIONES).
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C CC DD
C
C C Y D = CODIGOS DE LAS AREAS A FUNDIR.
C
C ULTIMA TARJETA.
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT08F001.
C *****
C
C PRIMERA TARJETA.
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C POR CADA LINEA.
C
C SEGUNDA TARJETA Y SIGUIENTES: CODIGO DE AREA A LA DERECHA, CODIGO
C DE AREA A LA IZQUIERDA, NUMERO DE PUNTOS Y COORDENADAS X E Y DE LOS
C PUNTOS, SEGUN EL FORMATO DESCRITO.
C
C ULTIMA TARJETA.
C 1      10      20      30      40      50      70      80
COL.....|.....|.....|.....|.....|.....|.....|.....|
C /*
C*****
C*****
C*****
C
C
C COMMON/A/COORD(200,100),POLY(500,2)

```



```

COMMON/B/LAR(100),ICOD(2,100),NAR,NLIN
CHARACTER FMT(80)
READ (5,100) NLIN
100 FORMAT(I4)
READ(5,200)(FMT(I),I=1,80)
200 FORMAT(80A1)
DO 1 I=1,NLIN
C
C SE LEEN LAS COORDENADAS DE LOS PUNTOS DE LAS LINEAS Y LOS CODIGOS
C ASOCIADOS A ELLAS.
C
READ(8,FMT) ICOD(1,I),ICOD(2,I),LAR(I),(COORD(J,I),J=1,LAR(I))
1 CONTINUE
READ(5,200)(FMT(I),I=1,80)
C
C SE LEE EL NUMERO DE FUSIONES.
C
READ(5,101) NFUS
101 FORMAT(2I3)
DO 2 I=1,NFUS
C
C SE LEEN LOS CODIGOS DE LAS AREAS A FUNDIR.
C
READ(5,101) K,L
CALL POLIG2(K,L,LI)
C
C SE ESCRIBE EL CODIGO DE UN AREA, EL NUMERO DE PUNTOS QUE LA DESCRIBEN
C Y LAS COORDENADAS DE LOS PUNTOS.
C
WRITE(6,FMT)I,LI,(POLY(J,1),POLY(J,2),J=1,L)
2 CONTINUE
C
C .....
C
STOP
END

```

#### 2.2.4. Reconstrucción de contornos y jerarquía espacial. Dos alternativas.

Un problema que se presenta frecuentemente y que, por ello, conviene tener resuelto de forma automática, es la deducción de la configuración de las unidades de superior nivel en la jerarquía espacial, a partir de la correspondiente a las unidades de inferior nivel que las constituyen.

Pensemos, por ejemplo en la definición digital de un determinado mosaico administrativo a diversos niveles jerárquicos. Es evidente que la descripción del cuarteado administrativo a otros niveles de la jerarquía está incluida en la de las unidades elementales.

Se trata, sin duda, de resolver un problema similar al de la estructuración adecuada de una información estadística referida a las unidades espaciales elementales que permita trabajar con datos a diferentes niveles de agregación. En este caso, se pretende una identificación de qué unidades están incluidas dentro de una de superior nivel en la jerarquía espacial, para poder así conocer el valor de una determinada característica en ésta como resultado de la suma de los valores de la característica en aquéllas.

De igual manera, en el problema que estamos considerando es necesario conocer qué unidades forman parte de otra más extensa para, posteriormente, eliminar las fronteras internas al conjunto. Así la nueva zona quedaría descrita por los límites de las unidades elementales que forman parte de ella con otras unidades que no estén incluidas en la que se acaba de reconstruir.

Es necesario, por tanto, resolver dos problemas. El primero, identificación de las partes de la nueva unidad, se ha resuelto por la utilización de códigos jerárquicos, como ahora comentaremos. El segundo, el establecimiento de la descripción digital de la nueva unidad espacial, se ha resuelto por dos vías alternativas de construcción automática de envolventes, que contemplan dos posibles modos de digitización de las unidades elementales: programas POLIGS3 y POLIGS4.

La sección de recuperación de las unidades implicadas en la formación de la nueva unidad, aunque difiera en su plasmación concreta, tiene el mismo fundamento en los dos procedimientos.

Supongamos, para facilitar la comprensión del proceso, que nos encontramos ante un conjunto de unidades espaciales jerarquizadas en dos niveles y que asignamos a cada unidad elemental un código compuesto (a,b), tal que "a" apunta a la unidad espacial de segundo orden de la que forma parte, y "b" caracteriza inequívocamente a la unidad elemental distinguiéndola de todas las que, como ella, forman parte de la unidad "a".

Consecuentemente con lo que hemos señalado en el párrafo



terior, la recuperación de todas las unidades elementales que constituyen una determinada unidad de segundo orden -por ejemplo la 02-, es una tarea de lectura de códigos y de consideración de los elementos caracterizados por códigos del tipo (2,b), siendo b cualquier valor, ya que no afecta al proceso, en este caso.

Con esta premisa común, los programas POLIGS3 y POLIGS4, efectúan la reconstrucción de unidades de superior nivel en la jerarquía espacial, a partir de las descripciones de las unidades elementales.

#### 2.2.4.1. El programa POLIGS3

Este programa es un desarrollo más de los procedimientos de finidos en POLIGS1 para la reconstrucción de áreas a partir de sus líneas de frontera.

La base de datos original es el conjunto de todas las líneas de frontera entre las unidades más elementales, afectadas por un doble código (área a la derecha, área a la izquierda) que, en esta ocasión, tiene una expresión más compleja.

El programa contempla cinco niveles de agregación posibles. Por este motivo, tanto el código de área a la derecha, como el código de área a la izquierda de cada línea, es un código compuesto de cinco subcódigos.

El programa, tal como se encuentra estructurado actualmente, reconstruye todas las unidades espaciales de un determinado ámbito, en un determinado nivel de la jerarquía -el que decide el usuario-, en función de todas las líneas límite de las unidades elementales.

Propiamente la reconstrucción de cada área la efectúa la subrutina POLIG3, que a su vez articula las subrutinas INVERT, DEPURA y TEST. La función de las subrutinas INVERT y DEPURA ya ha sido descrita páginas arriba. La subrutina TEST es la que decide, en función de los códigos compuestos que afectan a cada línea de la base de datos, si una línea ha de ser considerada o no en la reconstrucción de una determinada área. Brevemente, la subrutina TEST selecciona las líneas que son frontera de esa unidad y elude todas sus divisiones interiores, que son la plasmación de otros niveles más desagregados de la jerarquía espacial.

Incluimos a continuación el listado del programa principal. Los textos de las subrutinas POLIG3 y TEST pueden encontrarse en el apéndice final.



```

C
C A = NUMERO DE REGIONES.
C B = NUMERO DE PROVINCIAS DE LA REGION (H).
C C = NUMERO DE MUNICIPIOS DE LA PROVINCIA (H,I).
C D = NUMERO DE DISTRITOS DEL MUNICIPIO (H,I,J).
C E = NUMERO DE SECCIONES DEL DISTRITO (H,I,J,K).
C F = NUMERO DE LINEAS DE PARTIDA.
C G = OPTATIVO DE AGREGACION ESPACIAL (VER LISTADO).
C H = REGION EN CUESTION.
C I = PROVINCIA EN CUESTION.
C J = MUNICIPIO EN CUESTION.
C K = DISTRITO EN CUESTION.
C
C TERCERA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C (2(212,13,12,13),13,3X,10F5,0(16F5,0))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS
C CODIGOS DE CADA LINEA, EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS.
C
C CUARTA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C (212,13,12,13,3X,13/(8F10,3))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A ESCRIBIR EN LA UNIDAD 6
C EL CODIGO DE CADA AREA EL NUMERO DE PUNTOS Y LAS COORDENADAS DE LOS
C PUNTOS QUE LA DESCRIBEN.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT08F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C POR CADA LINEA.
C
C SEGUNDA TARJETA Y SIGUIENTES: CODIGO DE AREA A LA DERECHA, CODIGO
C DE AREA A LA IZQUIERDA, NUMERO DE PUNTOS Y COORDENADAS X E Y DE LOS
C PUNTOS, SEGUN EL FORMATO DESCRITO.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C /*
C *****
C *****
C *****
C

```

```

COMMON/A/COORD(200,100),POLY(500,2)
COMMON/B/LAR(100),ICOD(2,100,5),JOTAS(5),NAR1,NAR2,NAR3,NAR4,
*NAR5,NLIN,ISW,LPOLY
CHARACTER FMT(80)
DATA INFOR/0/
READ (5,100) NAR1,NAR2,NAR3,NAR4,NAR5,NLIN,ISW,IR,IP,IM,ID
WRITE(6,200) NAR1,NAR2,NAR3,NAR4,NAR5,NLIN,ISW,IR,IP,IM,ID
100 FORMAT(20I4)
200 FORMAT(1X,20I4)
READ(5,300) (FMT(I),I=1,80)
300 FORMAT(80A1)

C
C LECTURA DE LAS LINEAS DIGITIZADAS, DE SUS LONGITUDES Y DE LOS CO-
C DIGOS ASOCIADOS A ELLAS.
C
DO 1 J=1,NLIN
READ(8,FMT) (ICOD(1,J,K),K=1,5), (ICOD(2,J,K),K=1,5),LAR(J),
*(COORD(I,J),I=1,LAR(J))
IF(INFOR.NE.0)
*WRITE(6,201) (ICOD(1,J,K),K=1,5), (ICOD(2,J,K),K=1,5),LAR(J),
*(COORD(I,J),I=1,LAR(J))
201 FORMAT(1X,2(2I2,13,12,13),13,3X,10F6.2/(31X,10F6.2))
1 CONTINUE
READ(5,300) (FMT(I),I=1,80)
GOTO(2,3,4,5,6),ISW
2 CONTINUE
C *****
C ISW=1. SE PIDE QUE SE DIBUJEN LOS CONTORNOS DE TODAS LAS REGIONES
C AUTONOMICAS.
DO 7 I=1,NAR1
JOTAS(1)=I
CALL POLIG3
WRITE(6,FMT) (POLY(J,1),POLY(J,2),J=1,LPOLY)
7 CONTINUE
STOP
C *****
3 CONTINUE
C *****
C ISW=2. SE PIDE QUE SE DIBUJEN LOS CONTORNOS DE LAS PROVINCIAS DE
C UNA DETERMINADA REGION AUTONOMICA (IR).
JOTAS(1)=IR
DO 8 I=1,NAR2
JOTAS(2)=I
CALL POLIG3
WRITE(6,FMT) (POLY(J,1),POLY(J,2),J=1,LPOLY)
8 CONTINUE
STOP
C *****
4 CONTINUE
C *****
C ISW=3. SE PIDE QUE SE DIBUJEN LOS CONTORNOS DE LOS MUNICIPIOS DE
C UNA DETERMINADA PROVINCIA (IR,IP).
JOTAS(1)=IR
JOTAS(2)=IP
DO 9 I=1,NAR3
JOTAS(3)=I
CALL POLIG3
WRITE(6,FMT) (POLY(J,1),POLY(J,2),J=1,LPOLY)

```

```

9 CONTINUE
STOP
C *****
5 CONTINUE
C *****
C ISW=4. SE PIDE QUE SE DIBUJEN LOS CONTORNOS DE LOS DISTRITOS MUNI-
C CIPALES DE UN DETERMINADO MUNICIPIO (IR,IP,IM).
JOTAS(1)=IR
JOTAS(2)=IP
JOTAS(3)=IM
DO 10 I=1,NAR4
JOTAS(4)=I
CALL POLIG3
WRITE(6,FMT)(POLY(J,1),POLY(J,2),J=1,LPOLY)
10 CONTINUE
STOP
C *****
6 CONTINUE
C *****
C ISW=5. SE PIDE QUE SE DIBUJEN LOS CONTORNOS DE LAS SECCIONES CEN-
C SALES DE UN DETERMINADO DISTRITO MUNICIPAL (IR,IP,IM,ID).
JOTAS(1)=IR
JOTAS(2)=IP
JOTAS(3)=IM
JOTAS(4)=ID
DO 11 I=1,NAR5
JOTAS(4)=I
CALL POLIG3
WRITE(6,FMT)(POLY(J,1),POLY(J,2),J=1,LPOLY)
11 CONTINUE
STOP
C *****
C *****
END

```

#### 2.2.4.2. El programa POLIGS4

Este programa, aunque realiza un tipo de tareas muy próximo al que efectúa POLIGS3, tiene una estructura completamente distinta. La razón de ello es que contempla un modo de digitización diferente.

Mientras POLIGS3 parte de las líneas de frontera entre áreas -acompañadas de un doble código-, POLIGS4 considera las áreas de finidas, cada una de ellas, por una poligonal cerrada, caracterizada por un solo código -el que corresponde al área en cuestión-.

El programa a que nos estamos refiriendo efectúa la reconstrucción de un área determinada -la que se corresponde con un código que ha de introducir el usuario- a partir de las poligonales cerradas que describen las áreas, de inferior nivel en la jerarquía espacial, que la constituyen.

La selección de las áreas a considerar, de entre todas las que constituyen la base de datos, se realiza investigando adecuadamente los códigos de identificación que las afectan.

Una vez seleccionadas las áreas que han de considerarse se inicia un procedimiento iterativo en el que en cada iteración se funden dos polígonos, siendo uno de ellos el resultado de fusiones anteriores. De esta manera, cuando todos los polígonos que describen las unidades espaciales de inferior nivel han sido considerados, se dispone de la descripción numérica de la resultante, o envoltente, de todos ellos.

El programa, mucho más complejo que POLIGS3, articula diferentes subrutinas, cada una de las cuales soluciona uno de los diversos problemas implicados en la tarea general. Sus nombres son CONTIG, TESTFE, CLOCK, TESTFA, NEWWAY, REINSC, NEWAI y OTHWAY, y sus textos se pueden encontrar en el apéndice final.

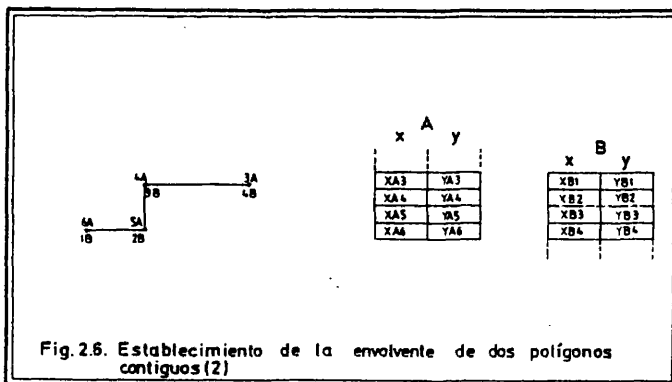
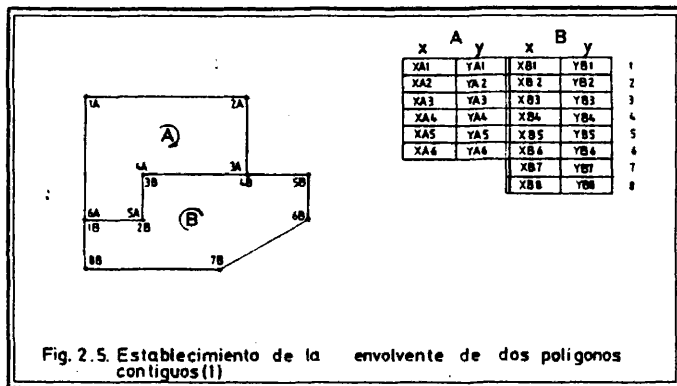
Vamos, a continuación, a explicar brevemente cómo se realiza la fusión de dos polígonos. Ya hemos dicho que la fusión de  $n$  polígonos se efectuará repitiendo  $n-1$  veces este procedimiento, considerando siempre el resultado de las fusiones anteriores y un nuevo polígono.

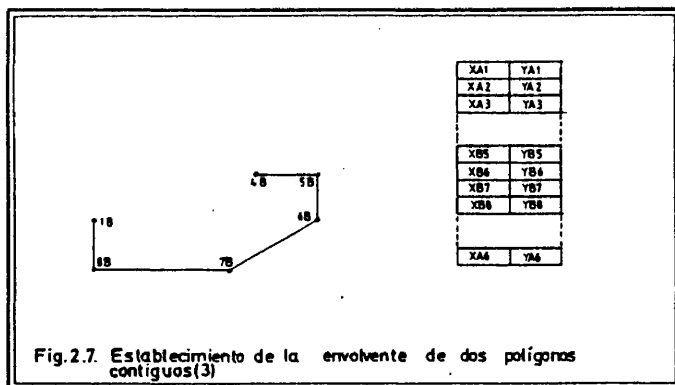
Supongamos una situación muy sencilla de dos polígonos descritos por las coordenadas de sus vértices, cuya envoltente queremos establecer (Fig. 2.5.).

El problema se resuelve identificando convenientemente en A el segmento que es común con B (Fig. 2.6.), y sustituyendo coherentemente esta sección en la descripción de A, por la descripción del resto del polígono B (Fig. 2.7.).

La búsqueda de un nuevo polígono que tenga frontera con el resultado de las fusiones anteriores está confiada a la subrutina CONTIG, así como la identificación de la frontera común.







La identificación coherente de la sección del nuevo polígono que no es común la realiza la subrutina NEWWAY.

La inserción adecuada de la sublista definida por NEWWAY en la lista que describe las fusiones anteriores la efectúa la subrutina REINSC.

Finalmente, las subrutinas TESFE, CLOCK, TESTFA, NEWAI y OTHWAY realizan manipulaciones a que no hemos hecho referencia, pero que son necesarias para el correcto funcionamiento del algoritmo de fusión. Un estudio detallado de los textos de todas las subrutinas que articula el programa POLIGS4 -que están ampliamente comentados en todas sus secciones- posibilitará la comprensión en profundidad del procedimiento.

Presentamos a continuación el texto de este programa, que ya ha sido utilizado en la definición digital del seccionamiento administrativo del municipio de Madrid.

```

C
C *****
C * PROGRAMA DE RECONSTRUCCION DE LIMITES DE AREAS DE SUPERIOR NI- *
C *VEL EN LA JERARQUIA ESPACIAL, A PARTIR DE SUS LIMITES DE LAS AREAS *
C *DE NIVEL INMEDIATAMENTE INFERIOR QUE LAS CONSTITUYEN. *
C * EL USUARIO HA DE DEFINIR EL CODIGO NUMERICO (INFC) QUE CONRES- *
C *PUNDE A LA UNIDAD ESPACIAL DE ORDEN SUPERIOR CUYOS LIMITES SE VAN *
C *A RECONSTRUIR. EL PROGRAMA RECURRE EL FICHERO DONDE SE ENCUENTRAN *
C *ALMACENADOS TODOS LOS LIMITES DE LAS UNIDADES DE ORDEN INMEDIATA- *
C *MENTE INFERIOR Y SULO CONSIDERA LOS LIMITES DE LAS QUE FURMAN PAR- *
C *TE DEL AREA EN CUESTION. *
C * EL PROGRAMA EDITA UN FICHERO EN LA UNIDAD 7, CON LA MISMA ES- *
C *TRUCTURA QUE EL DE LAS AREAS DE NIVEL INFERIOR: CODIGO DEL AREA, *
C *NUMERO DE PUNTOS QUE LA DEFINEN Y LISTA DE CUORDENADAS DE DICHS *
C *PUNTOS. *
C * EL PROGRAMA, EN SU ACTUAL VERSION, SULO PERMITE RECONSTRUIR UN *
C *AREA. SI SE DESEA, POR TANTO, RECONSTRUIR MAS DE UN AREA, ES NECE- *
C *SARIO UTILIZARLO TANTAS VECES COMO AREAS A RECONSTRUIR, DEFINIENDO *
C *EN CADA MOMENTO EL CODIGO NUMERICO DEL AREA EN CUESTION. *
C * EL PROGRAMA ASUME QUE LA DIGITIZACION DE LAS UNIDADES ESPACIA- *
C *LES DE INFERIOR NIVEL SE HA HECHO PUNTO POR PUNTO, HABIENDOSE DE- *
C *FINIDO ESTOS PREVIAMENTE SOBRE EL MAPA DE REFERENCIA. ESTO ES DE- *
C *BIDO A QUE EL PROGRAMA RECONOCE UNA FRONTERA COMUN A DOS POLIGONOS *
C *SI, Y SULO SI, ESTOS COMPARTEN DOS O MAS VERTICES. *
C * *****
C * EL PROGRAMA PUEDE CONSIDERAR HASTA 50 AREAS DE ORDEN INFERIOR, *
C *DEFINIDAS POR UN MAXIMO DE 100 PUNTOS, Y UN AREA RESULTANTE DEFI- *
C *NIDA POR UN MAXIMO DE 1000 PUNTOS. SI SE REBASAN ESTOS LIMITES ES *
C *NECESARIO REDIMENSIONAR LOS ARRAYS: P, E, LP, H, IDE. *
C *****
C
C NOMBRE DEL PROGRAMA: POLIGSA.
C
C LENGUAJE: FORTRAN77.
C
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C MEMORIA: 168 KBYTES.
C
C SUBROUTINAS NECESARIAS: CONTIG, TESTFE, CLUCK, TESTFA, NEWWAY,
C REINSC, NEWAI, DTHWAY.
C
C *****
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN DOS FICHEROS.
C
C FICHERO FIOSFOU1.
C *****
C

```

```

C PRIMEHA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA (PUR EJEMPLO).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C (313/(2F10,5))
C
C DESCRIPCION DEL FORMATO DE LECTURA EN EL FICHERO FT04F001 DE LOS CO-
C DIGOS DE LAS UNIDADES DE INFERIOR NIVEL, DE SUS LONGITUDES Y DE LOS
C VALORES DE LAS COORDENADAS DE LOS PUNTOS QUE LAS DEFINEN. (PUEDE SER
C CUALQUIERA).
C
C TERCERA TARJETA (PUR EJEMPLO).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C (213/(2F10,5))
C
C DESCRIPCION DEL FORMATO DE ESCRITURA EN LA UNIDAD 7 DEL CODIGO DEL
C AREA RECONSTRUIDA, DE SU LONGITUD Y DE LAS COORDENADAS DE LOS PUN-
C TOS QUE LA DEFINEN. (PUEDE SER CUALQUIERA).
C
C CUARTA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C AAAAAA B
C
C A = CODIGO DEL AREA A RECONSTRUIR.
C B = CONTROLADOR DE INFORMACION DEL PROGRAMA. SI B = 1 EL PROGRAMA
C FACILITA INFORMACION DEL PROCESO DE FUSION, A TRAVES DE LA IM-
C PRESORA DE LINEAS.
C
C QUINTA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C CCCC
C
C C = DISTANCIA MAXIMA QUE EL ORDENADOR UTILIZA PARA DECIDIR SI
C DOS VERTICES PERTENECIENTES A DOS POLIGONOS DISTINTOS SON REAL-
C MENTE EL MISMO.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT04F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT04F001 DD *
C
C SEGUNDA TARJETA (0 TARJETAS).
C

```

```

C COHERENTEMENTE CON EL FORMATO DESCRITO: CODIGOS DE AREA, LONGITUD,
C COORDENAS DE LOS PUNTOS QUE LA DEFINEN, CODIGOS DE AREA, LONGITUD,
C COORDENAS DE LOS PUNTOS QUE LA DEFINEN, CODIGOS DE AREA .....
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
C DL.....|.....|.....|.....|.....|.....|.....|
C /%
C
C *****
C *****
C *****
C
C COMMON/F/P(100,100),E(1000,2),A(50*2),DISMAX
C COMMON/G/LP(50),M(50),NP,LE,INFUR,OUT
C DIMENSION IDE(50)
C CHARACTER FMT(80),FMT1(80)
C
C INTEGER IUT,RECU,P,E,A,DISMAX
C LOGICAL NUMAS
C
C P = MATRIZ ENTERA EN LA QUE SE ALMACENAN LAS COORDENADAS DE LOS PUN-
C TOS QUE DEFINEN LOS LIMITES DE LAS UNIDADES ESPACIALES DE OR-
C DEN INFERIOR, QUE SE VAN A FUNDIR EN EL AREA 'IREC'.
C E = MATRIZ ENTERA EN LA QUE SE ALMACENA LA DESCRIPCION DE LA EN-
C VOLVENTE DE LOS POLIGONOS YA TRATADOS POR EL PROGRAMA, EN CA-
C DA PASO.
C A = MATRIZ ENTERA EN LA QUE SE ALMACENA EL SEGMENTO, QUE SE VA A
C INCLUIR EN LA ENVOLVENTE, DEL POLIGONO QUE EN CADA PASO SE RE-
C CONOCE COMO FRONTERIZO A ESTA.
C DISMAX = DISTANCIA MAXIMA ENTRE VERTICES IDENTICOS, QUE EL PROGRA-
C MA ADMITE.
C LP = VECTOR ENTERO DONDE SE ALMACENAN LOS NUMEROS DE PUNTOS QUE DE-
C FINEN LAS UNIDADES ESPACIALES DE ORDEN INFERIOR EN CONSIDERA-
C CION.
C M = VECTOR ENTERO QUE SE UTILIZA PARA CONOCER LAS UNIDADES ESPA-
C CIALES QUE YA HAN SIDO TENIDAS EN CUENTA EN ALGUNA ITERACION
C ANTERIOR DEL PROGRAMA.
C NP = NUMERO DE UNIDADES QUE SE VAN A FUNDIR EN LA UNIDAD SUPERIOR
C IREC.
C LE = NUMERO DE PUNTOS QUE DESCRIBEN LA ENVOLVENTE EN CADA PASO DEL
C PROGRAMA.
C INFUR Y OUT = VARIABLES DE CONTROL DE LA INFORMACION QUE OFRECE EL
C PROGRAMA.
C FMT = VECTOR DE CARACTERES PARA LA DESCRIPCION DEL FORMATO DE LECTU-
C RA DE LOS DATOS DE PARTIDA
C FMT1 = VECTOR DE CARACTERES PARA LA DESCRIPCION DEL FORMATO DE ES-
C CRITURA DE LA ENVOLVENTE EN LA UNIDAD 7.
C IDE = VECTOR ENTERO DONDE SE ALMACENAN LOS CODIGOS DE IDENTIFICA-
C CION DE LAS UNIDADES RECUPERADAS.
C
C *****
C
C INICIALIZACION DE LAS VARIABLES DE LECTURA Y ESCRITURA.
C
C IN = 5
C OUT = 6
C

```

```

C*****
C
C  LECTURA DE PARAMETROS.
C
C  FORMATO DE LECTURA.
C
C      READ(IN,100) FMT
C
C  FORMATO DE ESCRITURA.
C
C      READ(IN,100) FMT1
C      WRITE(OUT,101) FMT
C      WRITE(OUT,101) FMT1
C
C  CODIGO DE LA UNIDAD A RECONSTRUIR Y VARIABLE DE CONTRUL DE ESCRITURA
C
C      READ(IN,102) IREC,INFDR
C      WRITE(OUT,103) IREC,INFDR
C
C  DISTANCIA DE TOLERANCIA ENTRE VERTICES IDENTICOS.
C
C      READ(IN,104) DISMAX
C      WRITE(OUT,104) DISMAX
C*****
C
C  LECTURA, EN LA UNIDAD 4, DE TODAS LAS AREAS DE INFERIOR NIVEL Y SE-
C  LECCION DE LAS QUE FURMAN EL AREA 'IREC'.
C
C      IN=4
C      IP=0
C      K=0
C
C  1 CONTINUE
C      K=K+1
C      READ(IN,FMT,END=4) RECU,IDE(50),LP(50),(P(1,99),P(1,100),
C      *I=1,LP(50))
C      IF(IREC.NE.RECU) GOTO 3
C      NP=IP+1
C      NP2=NP+2
C      NP2H1=NP2-1
C      IDE(NP)=IDE(50)
C      LP(NP)=LP(50)
C      DO 2 I=1,LP(NP)
C      P(I,NP2H1)=P(I,99)
C      P(I,NP2)=P(I,100)
C  2 CONTINUE
C      IF(INFDR.EQ.1)
C      *WRITE(OUT,105) K,NP
C      IF(INFDR.EQ.1)
C      *WRITE(OUT,106) RECU,IDE(NP),LP(NP),(P(1,NP2H1),P(1,NP2),I=1,LP(NP))
C  3 CONTINUE
C      GOTO 1
C
C*****
C
C  INICIALIZACION DE VALORES PARA LA FORMACION DE LA ENVOLVENTE.
C

```

```

4 CONTINUE
C
C EL PRIMER POLIGONO EN CONSIDERACION CONSTITUYE LA ENVOLVENTE INICIAL
C
      LE=LP(1)
      DU 5 I=1,LE
      E (1,1)= P(1,1)
      E (1,2)= P(1,2)
5 CONTINUE
C
C EL PRIMER POLIGONO YA HA SIDO TRATADO, Y NINGUNO DE LOS DEMAS.
C
      M(1)=1
      DU 6 I=2,NP
      M(I)=0
6 CONTINUE
C
C*****
C*****
C
C FORMACION DE LA ENVOLVENTE.
C
      NOMAS=.FALSE.
      IPOL=0
7 CONTINUE
      IPOL=IPOL+1
      CALL CUNTIG(IP,11,12,13,14,NOMAS)
C
C HAY QUE FUNDIR EL POLIGONO IP, ENTRE SUS VERTICES 11 E 12, CON LA
C ENVOLVENTE ENTRE LOS VERTICES 13 E 14 DE ESTA.
C
      IF(NOMAS) GOTO 8
C
C TODAVIA HAY QUE SEGUIR FUNDIENDO POLIGONOS.
C
      CALL NEWWAY(IP,11,12,1R)
C
C EN LA MATRIZ A QUEDA ALMACENADA LA SECCION DEL POLIGONO IP QUE HAY
C QUE FUNDIR CON LA ENVOLVENTE.
C
      CALL REINSC(13,14,1R)
C
C SE HA FUNDIDO CON LA ENVOLVENTE.
C
      IF(INFOR.EQ.1)
      *WRITE(OUT,107)IDE(IP)
      IF(INFOR.EQ.1)
      *WRITE(OUT,FMT1)IHEC,LE,(E(1,1),E(1,2),I=1,LE)
      GOTO 7
C
C 8 CONTINUE
C
C SE HA FORMADO LA ENVOLVENTE DEFINITIVA.
C
      IF(INFOR.EQ.1)
      *WRITE(OUT,108) (M(I),I=1,NP)
      IF(INFOR.EQ.1)
      *WRITE(OUT,109) IPOL

```

```
C
C SE ECHIBE EN LA UNIDAD 7 LA ENVOLVENTE DEFINITIVA.
C
      OUT=7
      WRITE(OUT,FMT1)IREC,LE,(E(1,1),E(1,2),I=1,LE)
C
C*****
C*****
C
      STOP
100 FORMAT(80A1)
101 FORMAT(1X,80A1)
102 FORMAT(216)
103 FORMAT(1X,216)
104 FORMAT(1X,14)
105 FORMAT(' K = ',16,' NP = ',16)
106 FORMAT(316/(217))
107 FORMAT(16)
108 FORMAT(1514)
109 FORMAT('/ NU HA ENCONTRADO NINGUNA FRONTERA MAS, DESPUES DE ',14,'
      * ITERACIONES')
      END
```



#### 2.2.5. Verificación y corrección de trabajos de digitización de áreas.

Aunque se trata de un procedimiento auxiliar, quiero hacer referencia, en este momento, a la existencia de un programa de verificación, para la posterior corrección, de pruebas de digitización: MOSAICO.

Este programa realiza una operación muy simple, pero necesaria antes de aceptar definitivamente un determinado trabajo de digitización.

Dado que normalmente se han de realizar las digitizaciones -bien cuando se hacen manualmente, bien cuando se utiliza algún tipo de digitizador que no disponga de una pantalla interactiva- "a ciegas", es preciso reproducir sus resultados mediante algún terminal gráfico para comprobar su bondad.

El programa MOSAICO realiza dos operaciones muy sencillas: leer el número de puntos que describen una línea y las coordenadas de éstos, y dibujarla utilizando un plotter de pluma situando un identificador numérico sobre su centroide. Estas dos operaciones las realiza tantas veces como líneas existan en el fichero producto de una digitización.

Una vez obtenida la imagen correspondiente a una determinada digitización, se pueden corregir los valores de las coordenadas de los puntos que han sido digitizados erróneamente, bien manualmente, bien mediante algún procedimiento automático (1).

El examen del texto del programa facilita su comprensión y es necesario para conocer la documentación que registra acerca de sus requisitos de uso.

```

C
C *****
C * PROGRAMA DE DIBUJO DE CONTORNOS, DE UTILIDAD EN CORRECCION DE *
C * TRABAJOS DE DIGITIZACION. *
C * ***** *
C * SI ALGUNO DE LOS CONTORNOS ESTA DEFINIDO POR MAS DE 500 PUNTOS *
C * ES NECESARIO REDIMENSIONAR EL ARRAY XY. *
C *****
C
C NOMBRE DEL PROGRAMA: MOSAICO.
C
C LENGUAJE: FORTRAN77.
C
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C
C MEMORIA: 124 KBYTES.
C *****
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN DOS FICHEROS.
C
C FICHERO FT05F001.
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. |.....|.....|.....|.....|.....|.....|.....|
C AAA BBB CCC
C
C A = NUMERO DE CONTORNOS A DIBUJAR.
C B = FACTOR DE ESCALA DEL DIBUJO.
C C = ANGULO DE GIRO DEL IDENTIFICADOR NUMERICO QUE SE SITUA EN EL CEN
C TROIDE DE CADA CONTORNO.
C
C TERCERA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
C COL. |.....|.....|.....|.....|.....|.....|.....|
C (2F5.3)
C
C FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LAS COORDENADAS DE

```

```

C   LOS PUNTOS DE CADA CONTORNO.
C
C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   /*
C
C   FICHERO FT08F001.
C   *****
C
C   PRIMERA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   //FT08F001 DD *
C
C   POR CADA CONTORNO.
C
C   SEGUNDA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   AAA
C
C   A = NUMERO DE PUNTOS DEL CONTORNO.
C
C   TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y DE CADA UNO
C   DE LOS PUNTOS -POR ESTE ORDEN-, CON EL FORMATO DESCRITO.
C
C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   /*
C
C   *****
C   *****
C   *****
C
C   DIMENSION XY(2,500)
C   CHARACTER FMT(80)
C   INTEGER OUT
C   DATA IN,OUT/5,6/
C   READ (IN,100) NCONT,FCT,WR
C   WRITE (OUT,100) NCONT,FCT,WR
100  FORMAT (5X,13,2(5X,F3.0))
C   CALL PLOTS (0,0,6)
C   CALL FACTOR (FCT)
C   READ (IN,101) (FMT(I),I=1,80)
C   WRITE (OUT,102) (FMT(I),I=1,80)
101  FORMAT (80A1)
102  FORMAT ('OEL FORMATO DE LOS DATOS ES = ', 80A1)
C
C   SE SITUA EL ORIGEN DE COORDENADAS.
C
C   CALL PLOT(0.,0.,3)
C   CALL PLOT(3.,0.,2)
C   CALL SYMBOL(3.,-0.15,0.3,'X',0.,1)
C   CALL PLOT(0.,0.,3)
C   CALL PLOT(0.,5.,2)
C   CALL SYMBOL(-0.15,5.,0.3,'Y',0.,1)

```

```

      IN=8
      DO 1 I=1,NCNT
C
C   SE LEE EL NUMERO DE PUNTOS DEL CONTORNO.
      READ (IN,100) LCONT
      WRITE (OUT,100) LCONT
C
C   SE LEEN LOS PUNTOS DEL CONTORNO.
      READ (IN,FMT) ((XY(I1,J1),I1=1,2),J1=1,LCONT)
      WRITE (OUT,103) ((XY(I1,J1),I1=1,2),J1=1,LCONT)
103  FORMAT (8X,'X',9X,'Y',/ (5X,F6.2,4X,F6.2))
      IP=3
      N=LCONT+1
      XC=0.
      YC=0.
      DO 2 J=1,N
      X=XY(1,J)
      Y=XY(2,J)
      IF(J.E0.N) X=XY(1,1)
      IF(J.E0.N) Y=XY(2,1)
C
C   SE DIBUJA EL CONTORNO.
      CALL PLOT (X,Y,IP)
      IP=2
      XC=XC+X
      YC=YC+Y
      2  CONTINUE
      XC=XC/N
      YC=YC/N
      R1=FLOAT(1)
C
C   SE SITUA EL NUMERO DE ORDEN DEL CONTORNO EN EL INPUT, SOBRE EL CEN-
C   TROIDE DEL MISMO.
      CALL NUMBER(XC-0.1,YC-0.1,0.2,R1,WR,-1)
      1  CONTINUE
      CALL PLOT (-1.,-1.,999)
C
C.....
C
      STOP
      END

```

### 2.3. RAYADO DE POLIGONOS.

El rayado de polígonos es una tarea compleja que incluye todo el conjunto de manipulaciones y cálculos necesarios para el establecimiento de un haz de paralelas, de inclinación variable sobre la totalidad de un área definida por una poligonal cerrada.

Cuando se trabaja con gráficos vectoriales -como es el caso de todos los procedimientos que se presentan en esta tesis- el rayado de polígonos constituye la herramienta básica de reconocimiento automático de elementos superficiales -conjuntos de puntos interiores a una poligonal cerrada-.

En esta tesis se van a considerar dos de las principales aplicaciones que el rayado de polígonos tiene en el campo de la cartografía temática mediante ordenador. Dentro de este capítulo se presentan las posibilidades de los rayados en dos dimensiones en el establecimiento de mapas de coropletas, y en el siguiente, la utilización del rayado de polígonos en tres dimensiones en la definición numérica de bloques diagrama en perspectiva isométrica.

#### 2.3.1. Cálculo de los puntos de intersección de una línea recta con una poligonal cerrada. Solución de los problemas de límite y colinealidad.

El primer problema que hay que resolver, en términos generales, en el establecimiento del rayado de un polígono es el del cálculo de los puntos de intersección de una raya -línea recta- con una poligonal cerrada. Una vez resuelto este problema, el rayado se deriva de un procedimiento iterativo en el que, en cada iteración, se calculan y se unen adecuadamente mediante segmentos rectilíneos los puntos de corte de la poligonal con cada una de las rectas de un haz de paralelas.

Para solucionar este problema básico se ha diseñado una subrutina que lo resuelve automáticamente: RAY, que, a su vez, utiliza una subrutina REORG -para efectuar una determinada manipulación, necesaria en algunos supuestos- (2).

Esta subrutina, propiamente, calcula los puntos de corte de una poligonal definida por sus vértices con la recta  $Y = YR$ , paralela al eje de las X, y los devuelve ordenados en función de su coordenada X. YR es un parámetro externo a esta subrutina, que ha de definir en cada iteración el programa que la manipule.

El algoritmo de cálculo es muy eficaz, en el sentido de que incluye un test previo que determina qué segmentos del polígono interseca la raya. De esta manera el cálculo de intersecciones se limita únicamente a los casos reales del problema.

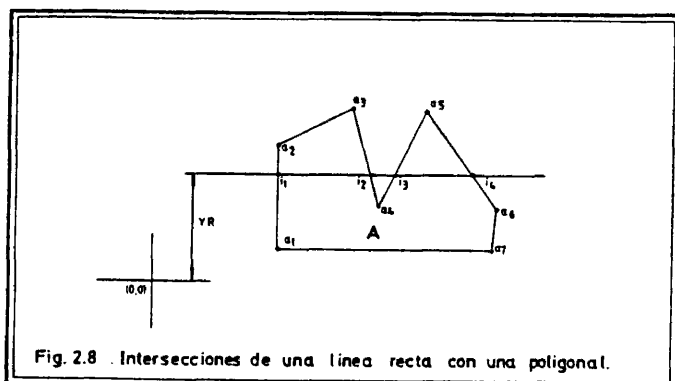
Una vez calculados todos los puntos de intersección -piénsese se que en el caso de polígonos cóncavos los puntos de intersección

pueden ser más de dos-, los ordena en función de su coordenada X. Esta operación es necesaria, porque de esta manera existe la seguridad de que, entre todos los segmentos definidos por los puntos de intersección, sólo son interiores al polígono los que tiene por origen un punto impar y por extremo un punto par.

Para resolver definitivamente el problema ha sido necesario considerar y solucionar automáticamente los casos en que uno, o varios, de los puntos de intersección coinciden con vértices de la poligonal -problema de límite y problemas de colinealidad de segmentos del polígono con la raya en cuestión-. Esto ha sido llevado a cabo por la inclusión de varios test previos a la consideración del punto de intersección y por la redacción de la subrutina REORG, que cambia el orden de la descripción por vértices de la poligonal en el caso de que se presenten problemas de colinealidad en el primer o en el último segmento de ésta.

Para el estudio más detallado del funcionamiento de la subrutina, se puede consultar el listado de ésta, así como el de la subrutina REORG, que aparecen, ampliamente comentados, en el apéndice final.

Vamos a considerar un ejemplo sencillo para ilustrar gráficamente el funcionamiento de este procedimiento. Supongamos un polígono definido por las coordenadas de sus vértices y una recta paralela al eje de las X (Fig. 2.8.).



La subrutina necesita conocer las listas de las coordenadas de los puntos que describen el polígono  $X = \{x_j, j = 1, 7\}$  e  $Y = \{y_j, j = 1, 7\}$  y el valor YR, que define la recta cuyos puntos de corte con la poligonal calcula.

La subrutina devuelve, como resultado de su funcionamiento, de lista  $XI = \{xi_k, k = 1,4\}$  que contiene, ordenadas, las cuatro coordenadas X de los cuatro puntos de corte.

Dado que la coordenada Y de todos los puntos es la misma: YR, éstos quedan así totalmente definidos. Además se comprueba que los segmentos  $i_1 i_2$  e  $i_3 i_4$  -cuyo origen es un punto de intersección impar y su extremo uno par- son interiores al polígono en cuestión.

Posteriormente, si se desea reproducir los segmentos interiores al polígono, basta desplazar la pluma de un plotter al punto de coordenadas  $(xi_1, YR)$ , sin pintar, y desplazarla luego pintando, al punto de coordenadas  $(xi_2, YR)$ , repitiendo la misma operación con los puntos  $i_3$  e  $i_4$ .

### 2.3.2. Intersecciones de una línea recta con diversas poligonales. Solución al problema de rayado de polígonos concéntricos.

En algún supuesto de rayado de polígonos, como veremos en el próximo capítulo en el desarrollo del programa PRISMAS, es necesario conocer los puntos de intersección de una única raya con diversos polígonos, para reconocer numéricamente qué segmentos de dicha raya son interiores a cada uno de los polígonos que corta.

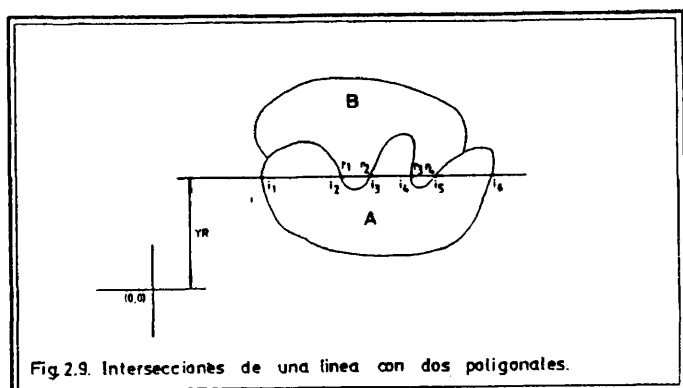
En el caso de que los polígonos no sean concéntricos, el problema, resuelto el que acabamos de comentar en el apartado anterior, no es complicado. Se trata, tan sólo, de construir una lista ordenada de tres columnas, en la que cada fila comprende los siguientes elementos: coordenada X del origen de un segmento interior al polígono  $i$ , coordenada X del extremo de este segmento, identificador numérico del polígono en cuestión. El criterio de ordenes tal que la magnitud de la coordenada X de los orígenes de todos los segmentos es creciente desde la fila 1 hasta la fila n, siendo n el número de segmentos definidos.

Ahora bien, si entre los polígonos que corta la raya se producen casos de concenticidad, el problema es bastante más complicado de resolver, ya que es necesario un tratamiento complejo de la lista a la que acabamos de hacer referencia en el párrafo anterior.

El problema ha sido resuelto diseñando dos subrutinas CHECK2 -que efectúa un test sobre la lista para ver si existen problemas de concenticidad- y CONCE2 -que resuelve el problema, fragmentando los segmentos interiores a polígonos que, a su vez, incluyen otros segmentos interiores a otros polígonos y reordenando la nueva lista-. Sus textos pueden encontrarse en el apéndice final. CONCE2 utiliza la subrutina ORDEN, cuyo listado aparece también en el apéndice, que es una subrutina genérica que ordena una matriz por filas.

Vamos ahora, con dos ejemplos gráficos sencillos a describir esquemáticamente el funcionamiento de estas subrutinas.

Supongamos que se trata de reconocer los segmentos interiores a dos polígonos que una misma raya paralela al eje de las  $\bar{X}$  determina (Fig. 2.9.).



La función -subrutina- RAY (A, YR) ha devuelto la lista  $XI = \{x_{ij}, j = 1,6\}$ , que determina tres segmentos interiores al polígono A:  $i_1i_2$ ,  $i_3i_4$  e  $i_5i_6$ .

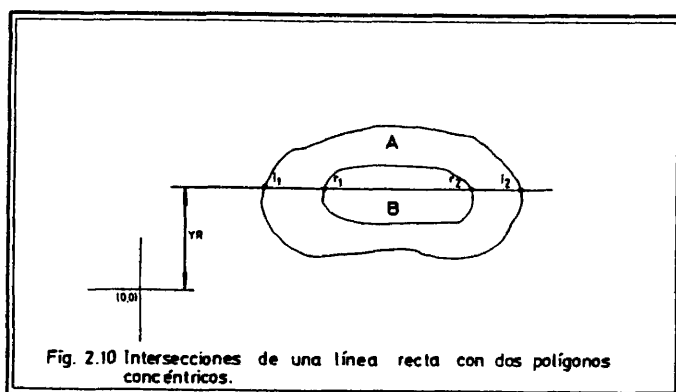
La función RAY (B, YR) devuelve, a su vez, la lista  $XR = \{x_{rj}, j = 1,4\}$ , que determina dos segmentos interiores al polígono B:  $r_1r_2$  y  $r_3r_4$ .

Ordenando los cinco segmentos por la coordenada X de su origen, se establece la siguiente lista que describe exactamente los segmentos interiores a A o a B que la tecta  $Y = YR$  determina.

$x_{i_1} \ x_{i_2} \ A$   
 $x_{r_1} \ x_{r_2} \ B$   
 $x_{i_3} \ x_{i_4} \ A$   
 $x_{r_3} \ x_{r_4} \ B$   
 $x_{i_5} \ x_{i_6} \ A$

Si consideramos, en cambio, la siguiente situación (Fig. 2.10.), el resultado de un tratamiento semejante al que acabamos de ver nos proporciona una lista que no describe correctamente los





segmentos interiores a A o a B que la recta  $y = YR$  determina.

```

xi1 xi2 A
xr1 xr2 B

```

Si denotamos genéricamente a esta lista por IBA, la función CONCE2 (CHECK2 (IBA)), dará como resultado otra lista que sí describe correctamente la situación real.

```

xi1 xr1 A
xr1 xr2 B
xr2 xi2 A

```

### 2.3.3. Definición de tramas simples.

Resuelto ya el problema de cálculo de los puntos de intersección de una recta paralela al eje de las X con una poligonal, no resulta complicado diseñar un procedimiento iterativo que calcule los puntos de corte de todas las rectas de un haz de paralelas a ella, separadas uniformemente entre sí.

Supongamos una situación sencilla como la que representa la figura 2.11.

Por sucesivas llamadas a la subrutina RAY, cambiando en cada caso el parámetro que define la recta de corte, se pueden calcular todos los puntos de intersección de este haz de paralelas

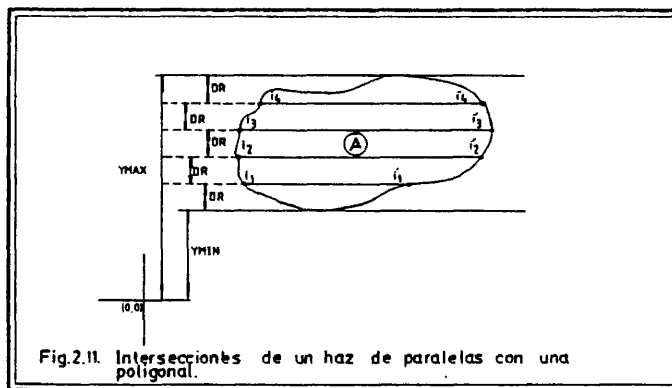


Fig.2.11. Intersecciones de un haz de paralelas con una poligonal.

con el polígono A, y reproducir posteriormente la trama de seada recurriendo a un periférico gráfico vectorial (plotter de pluma).

Más concretamente, el efecto deseado se obtendría por las siguientes llamadas a la subrutina RAY,

RAY (XY, (YMIN + DR), XC)	XY = Lista de todas las coordenadas de los puntos que definen el polígono.
RAY (XY, (YMIN + 2.DR), XC)	
RAY (XY, (YMIN + 3.DR), XC)	XC = Lista de coordenadas X de los puntos de corte que calcula la subrutina RAY.
RAY (XY, (YMIN + 4.DR), XC)	

que pueden expresarse de una forma más sintética de la siguiente manera:

/2.1/ (RAY (XY, (YMIN + DR.i), XC), i = 1,4)

Si deseamos obtener una trama más densa que la del ejemplo, bastaría con asignar al parámetro DR un valor menor.

Dado que YMIN e YMAX -mínimo y máximo valor de la lista de coordenadas Y de los puntos del polígono A- se pueden calcular automáticamente y que el número de rayas paralelas, separadas entre sí una distancia DR, que se pueden trazar entre YMIN e YMAX, es función precisamente de estos tres valores -YMIN, YMAX y DR-, se

comprende que la subrutina de tramado de paralelas horizontales no necesite más información que las listas de coordenadas X e Y del polígono a tramar y el valor que controla la densidad de la trama.

Si se acude al texto de la subrutina TRAMEL, en el apéndice final, se puede observar que sus parámetros son básicamente los que acabamos de señalar.

SUBROUTINE TRAMEL (XY, MAX, LXY, W, DR, INFOR)

XY = lista de LXY. 2 elementos que reúne las coordenadas del polígono a tramar.

MAX = longitud del array\* XY en el programa principal (no es relevante para lo que estamos ahora comentando).

LXY = número de puntos que definen el polígono.

DR = separación en cm. de las rayas de la trama.

INFOR = variable de control de información alfanumérica (no es relevante para lo que estamos ahora comentando).

Vemos que la subrutina TRAMEL tiene un parámetro W que todavía no hemos comentado y que tiene relación, como vamos a ver inmediatamente, con la obtención de tramas en las que el haz de líneas no es paralelo al eje de las X. Si prescindimos de él por un momento y de los parámetros MAX e INFOR, que no son relevantes, y suponemos conocida la longitud en filas de la lista XY, podemos establecer una definición bastante precisa de la subrutina TRAMEL, en función de la subrutina RAY,

/2.2/ TRAMEL (XY,DR)=(RAY(XY,(YMIN+DR.i),XC),DIB(XC),i=1,N)

en la que N es el número de rayas a trazar, que, como ya hemos visto hace un momento, es función de YMIN, YMAX y DR, y DIB (XC) es un procedimiento que une los puntos de corte calculados por RAY con segmentos rectilíneos que van desde punto impar a punto par.

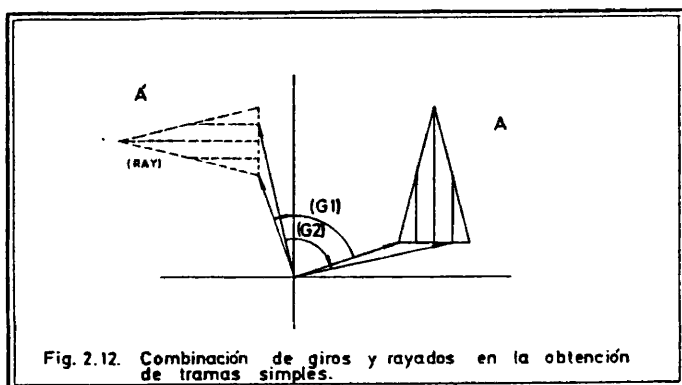
Para terminar ya este apartado dedicado a la definición de tramas simples, vamos a detenernos en el parámetro W de la subrutina TRAMEL para ver cómo funciona.

Hasta ahora hemos considerado en varios momentos que la subrutina RAY sólo puede calcular puntos de corte de una recta paralela al eje de las X con un polígono. Según esto, los procedimientos de tramado que se basen sólo en RAY, únicamente podrán obtener sombreados de intensidad variable constituidos siempre por paralelas al eje X más o menos próximas entre sí. Este hecho limita, sin duda, las posibilidades de definición de tramas. Por ello,

es necesario arbitrar algún sistema para reproducir tramas de inclinación también variable.

Para lograr el objetivo que hemos definido en el párrafo anterior caben dos alternativas. Una de ellas puede ser la definición de una nueva subrutina de rayado con diferentes ángulos. Otra, que es la que se ha adoptado (3), consiste en girar previamente un ángulo determinado el polígono a tramar, calcular los puntos de corte de ese nuevo polígono con un haz de paralelas horizontales -así puede seguirse recurriendo a la subrutina RAY- y, finalmente, antes de utilizarlos para el dibujo de los segmentos interiores al polígono, girar los puntos de corte del haz horizontal con el polígono un ángulo opuesto al interior. El ángulo de giro, y por tanto de la trama, es lo que se controla mediante el parámetro W de la subrutina TRAMEL. Para efectuar esta operación de giro se han diseñado las subrutinas antagónicas G1 y G2 -ver apéndice final-.

Mediante una figura sencilla se puede comprender el procedimiento (Fig. 2.12.).



De todo lo visto se deduce que la subrutina TRAMEL puede ser utilizada para reproducir tramas simples de cualquier densidad -controlada por el parámetro DR- e inclinación -controlada por el parámetro W-.

#### 2.3.4. Definición de tramas compuestas.

Antes de pasar a describir con detalle cómo se articula todo lo visto hasta el momento en un procedimiento de obtención automática de mapas de coropletas, me parece necesario detenerme en

una tarea que ha sido acometida para ampliar sustancialmente las posibilidades de definición de tramas mediante combinaciones de rayados.

Hasta ahora, las tramas reproducibles se derivan de la diversa inclinación y densidad que puede adoptar un haz de paralelas interiores al polígono en cuestión. Por la definición de una nueva subrutina -TRAMA- se pueden obtener tramas compuestas de diversos rayados -de inclinación o densidad diferente- sobre un mismo polígono.

Por el momento, se dispone de una minuta de 22 tramas que van desde la blanca -ausencia de rayado-, a la trama negra -rayado de intensidad máxima-, pero este repertorio se puede ampliar fácilmente en función de las necesidades que expresen los usuarios del programa que realiza automáticamente mapas de coropletas.

Utilizando la notación de /2.1/ y /2.2/ se puede definir, incluyendo el parámetro W, la subrutina TRAMA de la siguiente manera:

/2.3/ TRAMA (XY,I)=(TRAMEL(XY,DR(I,J),W(I,J)),J=1,N(I))

en donde,

XY = lista de coordenadas que definen el polígono a tramar.

I = identificador de la trama deseada (actualmente, un número entre 1 y 22).

N(I) = número de rayados necesarios para reproducir la trama I.

DR(I,J) = valor de la distancia entre líneas para reproducir la trama I, en cada uno de los sucesivos rayados.

W(I,J) = valor de la inclinación de las líneas para reproducir la trama I, en cada uno de los sucesivos rayados.

La minuta de las tramas, en su estado actual, puede consultarse en el listado del programa COROMAP, dentro de este capítulo, unas páginas más adelante.

## 2.4. ELABORACION AUTOMATICA DE MAPAS DE COROPLETAS.

### 2.4.1. Definición precisa de las tareas de acometer.

A la vista de todo lo anterior, y coherentemente con ello,

el establecimiento automático de un mapa de coropletas, en su sección gráfica, se reduce a la lectura de las descripciones digitales de un conjunto de polígonos geográficamente significativos y a su tramado posterior recurriendo a las subrutinas definidas previamente.

Sea  $XY$  la lista de las listas que describen todos los polígonos a tramar,  $XY = \{XY_1, XY_2, \dots, XY_n\}$ ,  $C$  la lista de los identificadores de trama que se corresponden con cada una de las sublistas de  $XY$ ,  $C = \{C_1, C_2, \dots, C_n\}$  y  $CIRC(XY_i)$  una rutina de dibujo de polígonos. El procedimiento iterativo  $MAP(XY, C)$ , de construcción automática de un mapa de coropletas, tendrá la siguiente estructura:

- MAP 1. Lee la siguiente lista  $XY_i = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_m, Y_m)\}$
- MAP 2. Lee  $C_i$
- MAP 3.  $CIRC(XY_i)$
- MAP 4.  $TRAMA(XY_i, C_i)$
- MAP 5. Si es la última lista, TERMINA, si no, vuelve a MAP 1.

Si nos hubiéramos limitado al problema gráfico exclusivamente, todo acabaría aquí. Esto exigiría del usuario del programa la clasificación previa de los casos a representar y la asignación de un mismo identificador de trama a todos aquellos que formaran parte del mismo grupo.

Hemos preferido, en cambio, facilitar al usuario un conjunto de rutinas de clasificación automática de los casos del problema, a partir de los valores de una o más características sobre cada uno de éstos.

La variedad de estos procedimientos de clasificación se explicita en el siguiente apartado. Ahora vamos a denominarles genéricamente  $CLASS(P, C)$ , siendo  $P$  la lista de los valores de una característica sobre  $n$  contornos  $P = \{P_1, P_2, \dots, P_n\}$ . En todos los casos, los diversos procesos de clasificación dan como resultado una lista  $C = \{C_1, C_2, \dots, C_n\}$ , cuyos elementos son identificadores de trama, tal que hace corresponder un mismo identificador a todos los casos de un mismo grupo de la clasificación.

Si redefinimos ahora el procedimiento  $MAP(XY, C)$ , como  $MAP2(XY, C)$  de la siguiente manera,

MAP2 (XY, C)

- MAP2 1. Lee la siguiente lista  $XY_i = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_m, Y_m)\}$

MAP2 2. CIRC ( $XY_i$ )

MAP2 3. TRAMA ( $XY_i, C_i$ )

MAP2 4. Si es la última lista, TERMINA, si no, vuelve a MAP2 1.

podemos definir en términos precisos el procedimiento

COROMAP ( $XY, P$ )

COROMAP1. Lee la lista  $P = \{ P_1, P_2, \dots, P_n \}$

COROMAP2. CLASS ( $P, C$ )

COROMAP3. MAP2 ( $XY, C$ ).

Dicho de una forma más coloquial, el procedimiento automático de elaboración de mapas de coropletas que presentamos en este apartado tiene las siguientes secciones.

En primer lugar una sección de lectura de los datos de base, que, en este caso, comprende la lectura de los pesos de la característica o características, en función de las cuales se va a efectuar una clasificación de los casos del problema, y de ciertos parámetros de control -sobre el modo de clasificación elegido y sobre la forma en que se va a reproducir el dibujo- que más adelante detallaremos.

A continuación, un conjunto flexible de cinco rutinas alternativas de clasificación.

En tercer lugar, una parte de lectura de las coordenadas que describen los contornos de las unidades espaciales y de tramado coherente de las mismas en función de la clasificación previa.

Finalmente, una sección optativa a la que se puede recurrir para la obtención de una información gráfica adicional, necesaria normalmente para la interpretación adecuada del gráfico -diagrama tramado de frecuencias, leyenda de tramas y rótulo explicativo del mapa resultante-.

#### 2.4.2. Problemas de clasificación que están relacionados con la construcción de un mapa de coropletas

Los mapas de coropletas representan siempre la componente espacial de una clasificación previa de las unidades de referencia, realizada en función de alguna característica estructural.

Por este motivo, un programa de elaboración automática de mapas de coropletas debe ofrecer siempre un conjunto de subrutinas de clasificación, lo más flexible, de los casos del problema.

El programa que presentamos ofrece al usuario cinco vías alternativas de clasificación de los casos del problema: clasificación standard en seis intervalos de igual amplitud, clasificación por centiles, clasificación por unidades de desviación típica, clasificación por umbrales previamente definidos y cluster de proximidad máxima a partir de valores de hasta diez características sobre los casos, o, también, a partir de una matriz de distancias intercasos, calculada con antelación.

Excepto en el caso de que el usuario opte por el procedimiento de clasificación standard, siempre se puede decidir sobre el número de intervalos de la clasificación -con un tope máximo de 10, ya que superar este límite dificulta claramente la legibilidad del mapa resultante-.

Llegados a este punto, resulta imprescindible hacer un serie de puntualizaciones sobre la sección de clasificación del programa que presentamos.

En primer lugar, señalar que no vamos a comentar en detalle ninguna de las rutinas de clasificación. Esto es así por dos motivos. Por lo que respecta a la descripción de los algoritmos implementados, porque los listados de las subrutinas se encuentran suficientemente comentados en sus partes fundamentales como para ser comprendidos directamente -ver apéndice final, subrutinas CLASST, CLASCT, CLASDT, CLASUM y CLUSTR-. Por lo que respecta al sentido e interpretación de los diversos procedimientos de clasificación, porque éstos son suficientemente conocidos y porque, además, se puede obtener una descripción apropiada de los mismos en cualquier manual de estadística aplicada. En este sentido, solamente comentar que los procedimientos de clasificación standard, por centiles, por unidades de desviación típica y por umbrales se refieren a situaciones en las que se pretende establecer una clasificación en función de una sola característica y que el procedimiento de cluster puede ser utilizado para efectuar una clasificación en función de varias características ortogonales -esto es así porque calcula una matriz de distancias euclídeas- o en función de una matriz de distancias intercasos previamente calculadas.

En segundo término, afirmar que no se ha realizado ninguna discusión en profundidad, de la adecuación de los procedimientos de clasificación que se han formulado, a supuestos de cartografía mediante coropletas. La justificación de ello es que se trata de algo, si bien realmente interesante, ajeno al objetivo fundamental de esta tesis, que tiene una orientación claramente gráfica y cartográfica. Por este motivo se ha optado por ofrecer una amplia gama de posibilidades y cubrir así un mayor número de necesidades, en espera de que por la experimentación de todas ellas se pueda, con el tiempo, llegar, quizá, a la supresión de alguna y a la formulación de nuevas alternativas más apropiadas.

Finalmente, a pesar de lo que se ha dicho en el párrafo anterior, pensamos que la enorme facilidad en la obtención de una



respuesta gráfica -plasmación de la componente espacial-, coherente con un determinado supuesto de clasificación, posibilita la realización de diversos ensayos de cartografía hasta lograr la confección del mapa óptimo. Este hecho supone, sin duda, una ventaja que compensa, esa es nuestra opinión, la oferta indiscriminada de procedimientos de clasificación no experimentados y criticados en profundidad.

#### 2.4.3. El programa COROMAP

En el apartado que hemos dedicado a la definición en términos precisos de las tareas necesarias para el establecimiento de un procedimiento automático de confección de mapas de coropletas ya hemos dado una descripción sumaria de la estructura y funciones del programa COROMAP. Vamos en este apartado a detallar más los términos de la presentación y a mostrar algunos ejemplos de los resultados que se pueden obtener mediante su uso.

##### 2.4.3.1. Articulación de las rutinas de clasificación y de dibujo.

El programa COROMAP es una articulación de subrutinas jerarquizadas, que dan como resultado la confección de un mapa de coropletas en un plotter de pluma -CALCOMP 960-.

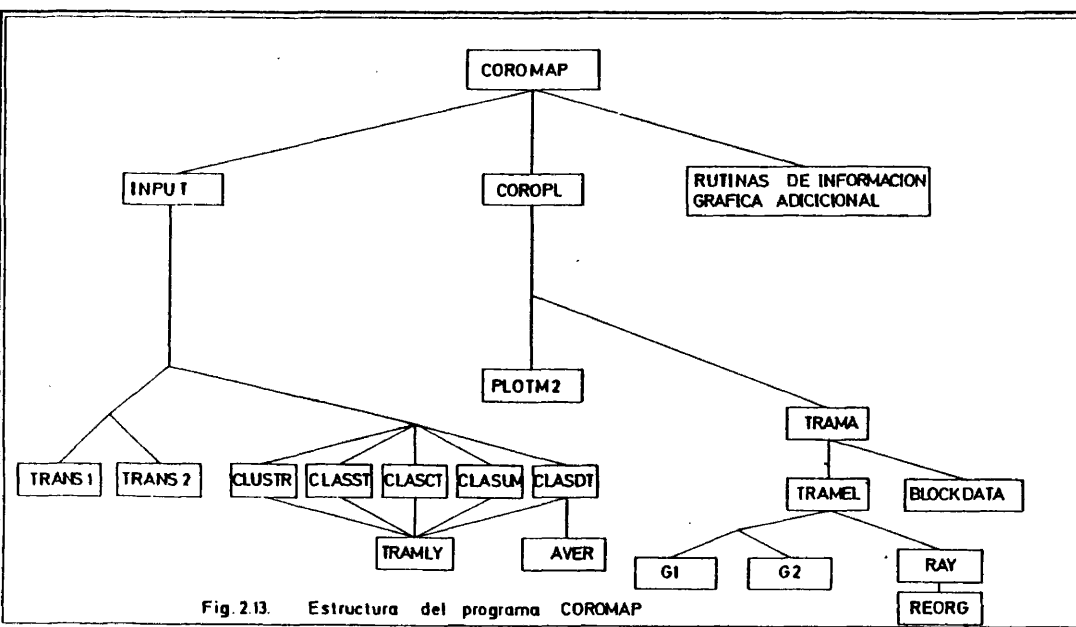
El programa principal, recurre directamente a seis subprogramas, de muy diferente complejidad, que a su vez recurren a toda una cadena de subrutinas que realizan separadamente diversas funciones.

La confección del mapa temático la realizan propiamente dos de los seis subprogramas a que se ha hecho referencia: INPUT y COROPL. Los cuatro restantes reproducen información gráfica adicional y a ellos se dedica un subapartado unas páginas más adelante.

Del siguiente árbol de dependencias se deriva una imagen esquemática de la estructura del programa COROMAP, tal como ha sido diseñado (Fig. 2.13.).

El subprograma INPUT tiene asignadas como funciones fundamentales, las siguientes:

- Lectura de la casi totalidad de los parámetros de control del programa y de los valores de la característica o características que van a determinar la clasificación de los casos. En esta tarea puede recurrir en algún momento a las subrutinas TRANS1 y TRANS2.
- Clasificación de los casos del problema. Esta tarea la efectúa recurriendo, de acuerdo con la petición del usuario, a una de las siguientes subrutinas: CLUSTR, CLASST, CLASCT, CLASUM o CLASDT -que a su vez invoca a la función AVER-.



- Devolución al programa principal de la lista de identificadores de trama correspondientes a cada caso, coherente con la clasificación resultante.
- Devolución al programa principal -a través de la subrutina TRAMLY- de una información numérica necesaria para la reproducción de la información gráfica adicional que el programa oferta.

El subprograma COROPL realiza las siguientes tareas:

- Lectura de los contornos de las áreas a tramar.
- Dibujo de los contornos. En esta sección recurre a la sub

rutina PLOTM2, que a su vez recurre a rutinas del software básico del sistema CALCOMP.

- Tramado coherente de los contornos acorde con la lista de identificadores de trama que se crea en INPUT.

En esta tarea se recurre a la subrutina TRAMA y a todas las que dependen de ella en el árbol de la figura 2.13.

Una de ellas, TRAMEL, recurre a rutinas del software básico del sistema CALCOMP.

#### 2.4.3.2. Información gráfica adicional que proporciona.

Los restantes cuatro programas a los que recurre directamente el programa COROMAP, realizan funciones de ilustración gráfica adicional.

El subprograma DIAGRM utiliza la información preparada por la subrutina TRAMLY y confecciona un diagrama de frecuencias tramado. Para ello recurre a la subrutina TRAMA.

El subprograma LEGEND reproduce la leyenda explicativa de las tramas -umbrales inferior y superior de cada grupo especificado en el mapa por una trama común-. Para ello necesita recurrir también a la información preparada por TRAMLY y a las prestaciones de las subrutinas TRAMA y CUADRA.

El subprograma TITULO recoge un rótulo de 240 caracteres máximo, introducido por el usuario, y lo reproduce en la cabecera del mapa.

Finalmente, el subprograma MARCO dibuja un rectángulo que enmarca toda la imagen resultante.

#### 2.4.3.3. Listado del programa y requisitos de uso del mismo.

Reproducimos a continuación el texto, documentado, del programa COROMAP, en el que aparecen claramente explicitados todos sus requisitos de uso.

Los textos de todas las subrutinas que utiliza se encuentran en el apéndice final, clasificados por orden alfabético.

```

C
C *****
C * PROGRAMA DE DIBUJO DE MAPAS TEMATICOS, MEDIANTE COROPLETAS (TRA*
C *MAS SIGNIFICATIVAS EN FUNCION DEL PESO DE UNA VARIABLE EN DIVERSAS*
C *UNIDADES ESPACIALES (1) DEFINIDAS POR SUS CONTORNOS). *
C * *****
C * (1) EL PROGRAMA PUEDE CONSIDERAR HASTA UN MAXIMO DE 100 AREAS DIS*
C *TINTAS. SI SE QUIERE UTILIZAR ESTE PROGRAMA MAS ALLA DE ESTE LIMI*
C *TE, ES NECESARIO REDIMENSIONAR LOS SIGUIENTES ARRAYS: PESO, PESOS,*
C *ITRAM, IA Y D. *
C *****
C *****
C
C NOMBRE DEL PROGRAMA: CROMAP.
C
C LENGUAJE: FORTRAN 77.
C
C AUTDR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C
C MEMORIA: 264 KBYTES.
C
C SUBRUTINAS NECESARIAS: INPUT, TRANS1, TRANS2, CLASST, CLASUM, CLASCT
C CLASDT, AVER(FUNCTION), CLUSTR, TRAMLY, CROPL, PLOTM2, TITULO,
C DIAGRM, LEGEND, CUADRA, TRAMA, TRAMEL, RAY, REORG, G1, G2, BLOCKDATA
C Y MARCO.
C
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN TRES FICHEROS.
C
C FICHERO FT05F001 (REUNE TODOS LOS PARAMETROS DE CONTROL).
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
C COL |.....|.....|.....|.....|.....|.....|.....|
C A
C A = OPTATIVO QUE DEFINE SI SE OPTA POR EL PROCEDIMIENTO STANDARD
C DE CLASIFICACION (A = 1), O POR OTRO PROCEDIMIENTO (A = 2).

```

```

C
C TERCERA TARJETA (OPCION STANDARD).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C
C B
C B = OPTATIVO QUE PERMITE LEER CON FORMATO PROPIO.
C B = 1. SE LEEN LOS DATOS CON FORMATO STANDARD.
C B = 2. SE LEEN CON EL FORMATO QUE DECIDE EL USUARIO.
C
C CUARTA TARJETA (OPCION STANDARD). SOLO APARECE SI B = 2.
C (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C (15/(F10.0))
C
C DESCRIPCION DEL FORMATO CON EL QUE SE HAN DE LEER EN LA UNIDAD 7 LOS
C SIGUIENTES DATOS: NUMERO DE AREAS Y VALOR DE LA CARACTERISTICA EN
C CADA UNA DE ELLAS.
C
C TERCERA TARJETA (OPCION NO STANDARD).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C C DD E F G
C
C C = OPTATIVO DE PROCEDIMIENTO DE CLASIFICACION.
C C = 1. CLASIFICACION SEGUN UMBRALES QUE APORTA EL USUARIO.
C C = 2. CLASIFICACION POR CENTILES.
C C = 3. CLASIFICACION POR UNIDADES DE DESVIACION STANDARD.
C C = 4. CLUSTER.
C C = 5. CLUSTER, A PARTIR DE MATRIZ DE DISTANCIAS.
C DD = NUMERO DE INTERVALOS DE CLASE QUE SE DESEAN (NUNCA MAYOR QUE 10).
C E = OPTATIVO DE LECTURA STANDARD-NO STANDARD.
C E = 1. SE LEEN LOS DATOS CON FORMATO STANDARD.
C E = 2. SE LEEN LOS DATOS CON EL FORMATO QUE DECIDE EL USUARIO.
C F = OPTATIVO QUE PERMITE EFECTUAR LECTURAS Y TRANSFORMACIONES PRE-
C VIAS DE LOS DATOS DE ENTRADA, QUE HA DE DEFINIR EL USUARIO.
C F = 1. SE LEEN LOS DATOS EN INPUT.
C F = 2. SE LEEN LOS DATOS EN TRANS1 O EN TRANS2 (ESTO PERMITE
C LECTURAS ESPECIALES Y TRANSFORMAR LOS PESOS DE LA VARIABLE,
C SI SE DEFINEN ADECUADAMENTE LAS SUBROUTINAS TRANS1 O TRANS2).
C G = OPTATIVO DE AMPLITUD DE CLASE EN EL PROCEDIMIENTO DE CLASIFICA-
C CION POR UNIDADES DE DESVIACION STANDARD.
C G = 1. EN LA CLASIFICACION POR UNIDADES DE DESVIACION STANDARD
C LA AMPLITUD DE LAS CLASES ES DE UNA DESVIACION STANDARD.
C G = 2. EN LA CLASIFICACION POR UNIDADES DE DESVIACION STANDARD
C LA AMPLITUD DE LAS CLASES ES DE 1/2 DESVIACION STANDARD.
C
C CUARTA TARJETA (OPCION NO STANDARD).
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C HH II JJ KK LL MM NN OO PP QQ
C
C LECTURA DE LAS TRAMAS ASOCIADAS A CADA INTERVALO DE CLASE. EN ESTA
C TARJETA DEBEN APARECER TANTOS NUMEROS COMO INTERVALOS DE CLASE HA-
C YA DEFINIDO PREVIAMENTE EL USUARIO. ESTOS NUMEROS, QUE PUEDEN OSCI-
C LAR ENTRE EL 1 Y EL 22 CAUSAN QUE SOBRE TODOS LOS CONTORNOS CORRES-
C PONDIENTES AL PRIMER INTERVALO DE CLASE SE SITUE LA TRAMA HH, SOBRE

```



```

C   RRR SS
C
C   RRR = NUMERO DE CASOS EN EL CLUSTER.
C   SS = NUMERO DE VARIABLES EN EL CLUSTER(MAXIMO 10).
C
C   SEXTA TARJETA (PROCEDIMIENTO NO STANDARD Y C = 4 O 5). SI E = 2
C   (COMO EJEMPLO).
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   (15/(F10.0))
C
C   DESCRIPCION DEL FORMATO CON EL QUE SE HAN DE LEER EN LA UNIDAD 7 LOS
C   PESOS DE LAS SS VARIABLES SOBRE LAS RRR AREAS.
C
C   SEPTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C           TTT      UUU      VVV
C
C   TTT = MARGEN LATERAL DEL DIBUJO RESPECTO DEL MARCO (EN CMS).
C   UUU = DIMENSION VERTICAL (EN CMS) A LA QUE SE AJUSTA EL DIBUJO DEL
C   DIAGRAMA DE FRECUENCIAS Y LA LEYENDA.
C   VVV = FACTOR DE ESCALA DEL DIBUJO.
C
C   OCTAVA TARJETA (COMO EJEMPLO).
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   (2F10.0)
C
C   DESCRIPCION DEL FORMATO CON EL QUE SE HAN DE LEER EN LA UNIDAD 8 LAS
C   COORDENADAS DE LOS CONTORNOS.
C
C   NOVENA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   X
C   X = NUMERO DE TARJETAS (MAXIMO 3) QUE FORMAN EL TITULO DEL DIBUJO.
C
C   DECIMA TARJETA (O TARJETAS).
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   MAPA DE DISTRIBUCION DE.....
C
C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   /*
C
C   FICHERO FT07F001 (NUMERO DE AREAS Y VALOR DE LA CARACTERISTICA
C   ***** EN ELLAS).
C
C   PRIMERA TARJETA.
C   1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
C   //FT07F001 DD *
C
C   FORMATO STANDARD.
C   *****

```

```

C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C AAA
C
C AAA = NUMERO DE AREAS.
C
C TERCERA TARJETA Y SIGUIENTES.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C BBBBBBBB CCCCCCCC DDDDDDDD EEEEEEEE FFFFFFFF GGGGGGGGG
C
C VALORES DE LA CARACTERÍSTICA EN LAS DIFERENTES AREAS.
C
C FORMATO NO STANDARD.
C *****
C LA INFORMACION CORRESPONDIENTE A ESTE FICHERO SE ORGANIZARA CDHEREN-
C TEMENTE CON EL FORMATO DESCRITO.
C
C NOTA : LA ENTRADA DE DATOS AL CLUSTER TIENE ALGUNA PECULIARIDAD.
C **** (VER INPUT).
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT08F001 (COORDENADAS DE LOS CONTORNOS).
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C POR CADA CONTORNO.
C *****
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C AA
C
C AA = NUMERO DE PUNTOS QUE DESCRIBEN EL CONTORNO.
C
C A CONTINUACION, SOBRE LAS TARJETAS QUE SEAN NECESARIAS Y CDHERENTE-
C MENTE CON EL FORMATO DESCRITO, COORDENADAS X E Y DE LOS PUNTOS.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C *****
C *****
C *****
C
C COMMON/XYHM/XTOP,YTOP,XBOTT,YBOTT
C
C EN ESTA ZONA COMUN SE ALMACENAN LOS VALORES DE LOS VERTICES DEL MI-

```



```

C NIMO RECTANGULO QUE ENMARCA EL DIBUJO DEL MAPA.
C .....
C INICIALIZACION DE VALORES.
C
C   XTOP= -1.E74
C   YTOP= -1.E74
C   XBOTT= 1.E74
C   YBOTT= 1.E74
C .....
C LLAMADA A LA SUBROUTINA DE LECTURA DE LOS PESOS DE LA VARIABLE SOBRE
C LAS N AREAS DEL PROBLEMA, Y, A TRAVES DE ELLA, A LAS SUBROUTINAS DE
C CLASIFICACION POR LAS QUE SE PUEDE OPTAR.
C
C   CALL INPUT(N,ILEY)
C .....
C SECCION DE DIBUJO.
C SE ABRE EL DIBUJO.
C
C   CALL PLOTS(0,0,6)
C
C LECTURA DE PARAMETROS DEL DIBUJO: MARGEN LATERAL DEL DIBUJO RESPECTO
C DEL MARCO; DIMENSION VERTICAL A LA QUE SE AJUSTARAN EL DIAGRAMA DE
C FRECUENCIAS Y LA LEYENDA; FACTOR DE ESCALA.
C
C   READ(5,100) XOR,YOR,FCT
C 100 FORMAT(5X,3F10,2)
C   CALL PLOT(XOR,YOR,-3)
C   CALL FACTOR(FCT)
C
C LLAMADA A LA SUBROUTINA DE DIBUJO DEL MAPA DE COROPLETAS.
C
C   CALL COROPL(N,FCT)
C
C LLAMADA A LAS SUBROUTINAS QUE DIBUJAN EL DIAGRAMA DE FRECUENCIAS,
C LA LEYENDA Y EL TITULO DEL MAPA.
C
C   XTOP=XTOP*FCT
C   YTOP=YTOP*FCT
C
C   CALL FACTOR(1.)
C
C   XDIAG=0.
C   CALL DIAGRM(XDIAG,-(YOR-1.),XTOP/2.,YOR-1.)
C   IF((ILEY,NE.4).AND.(ILEY,NE.5))
C     *CALL LEGEND(XTOP/2.,-(YOR-1.),XTOP/2.,YOR-1.)
C     CALL TITULO(XDIAG,YTOP,XTOP,YTOP/5.)
C
C SE ENMARCA EL DIBUJO.
C
C   CALL PLOT(-XOR,-YOR,-3)
C   XMARC=0.

```

```
YMARC=0,  
XD=2*XOR+XTOP  
YD=YOR+YTOP+YTOP/5.+1.  
CALL MARCO(XMARC,YMARC,XD,YD)
```

```
C  
C SE CIERRA EL DIBUJO.  
C
```

```
CALL PLOT(1.,1.,999)
```

```
C  
C .....  
C
```

```
STOP  
END
```

#### 2.4.3.4. Algunos ejemplos.

Hasta el momento, las únicas salidas gráficas de que se dispone, como ejemplo de las prestaciones del programa, son los siguientes mapas de coropletas de base provincial que representan valores de densidad de población en la España peninsular.

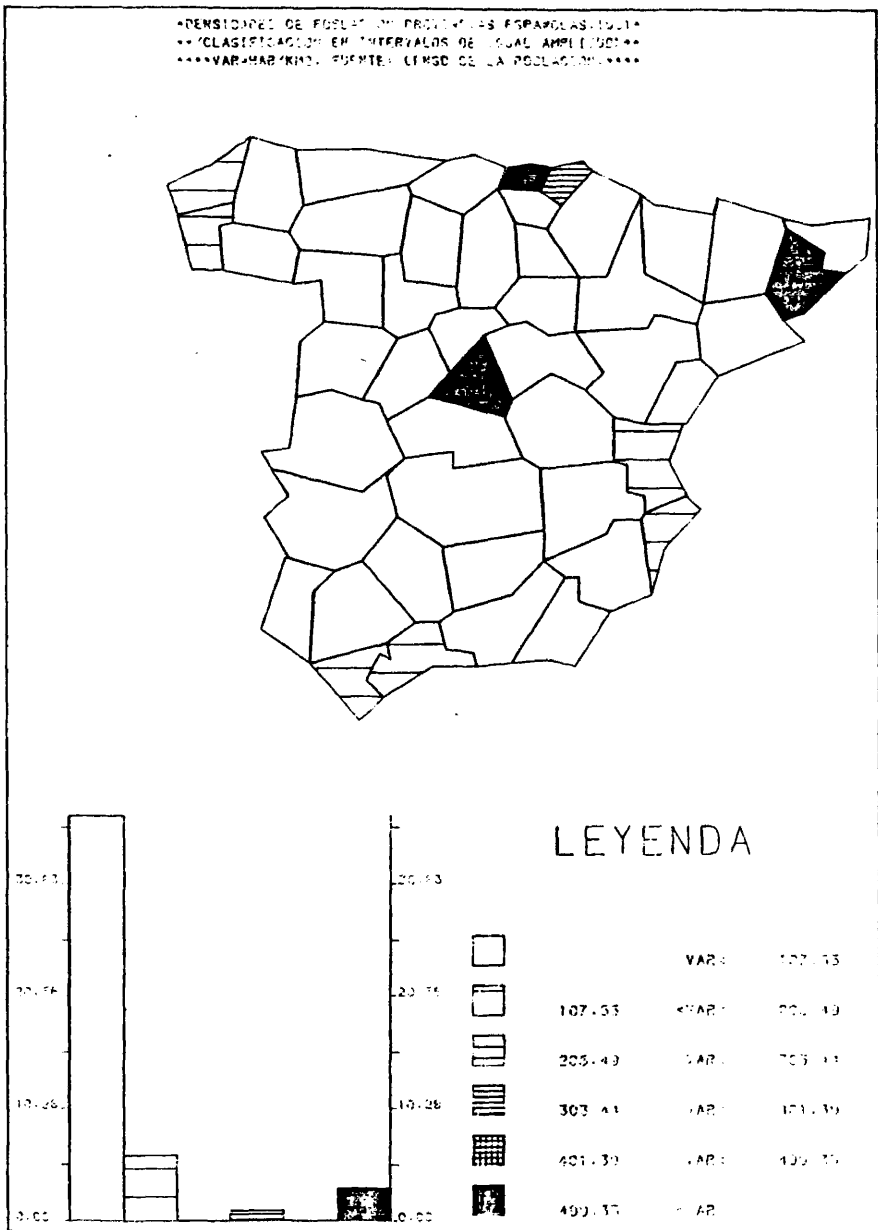


Fig. 2.14.



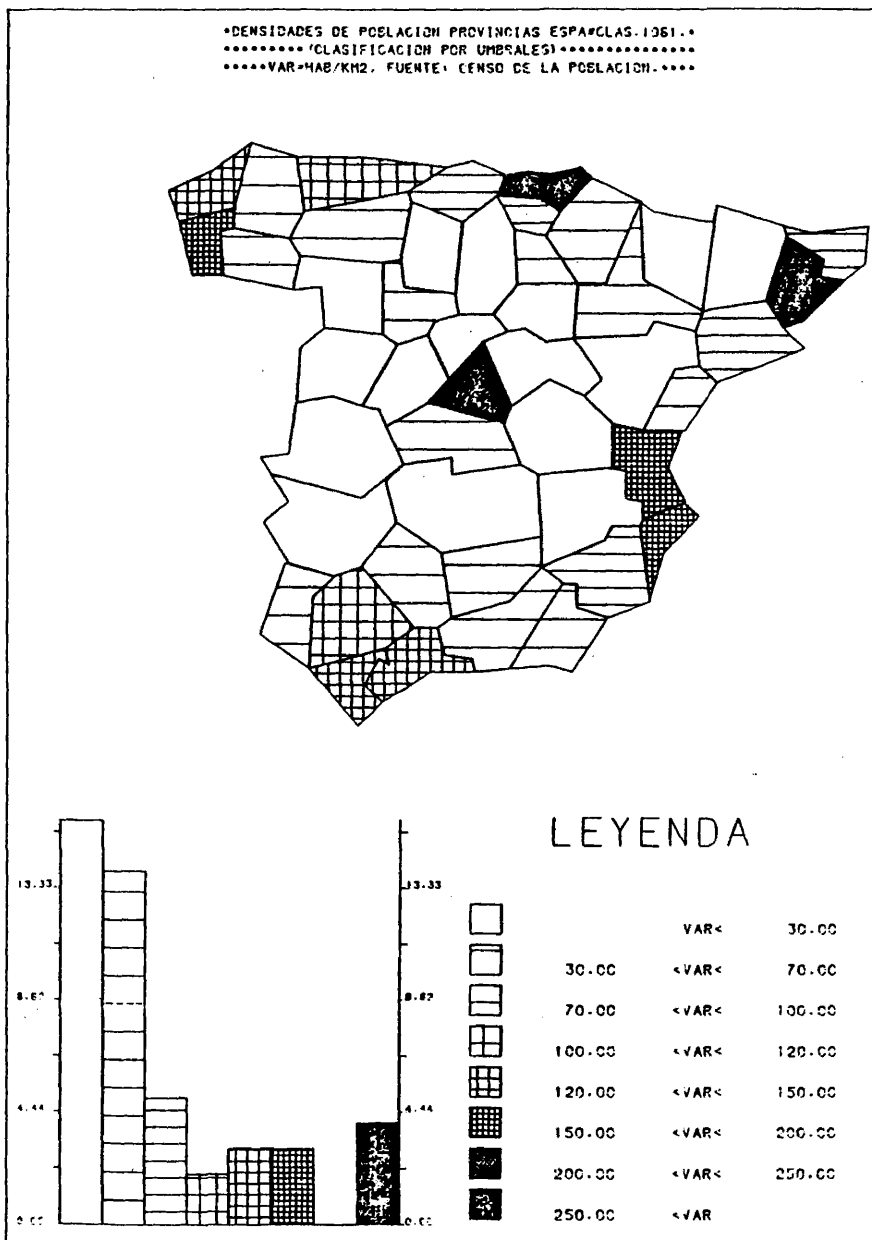


Fig. 2.16.

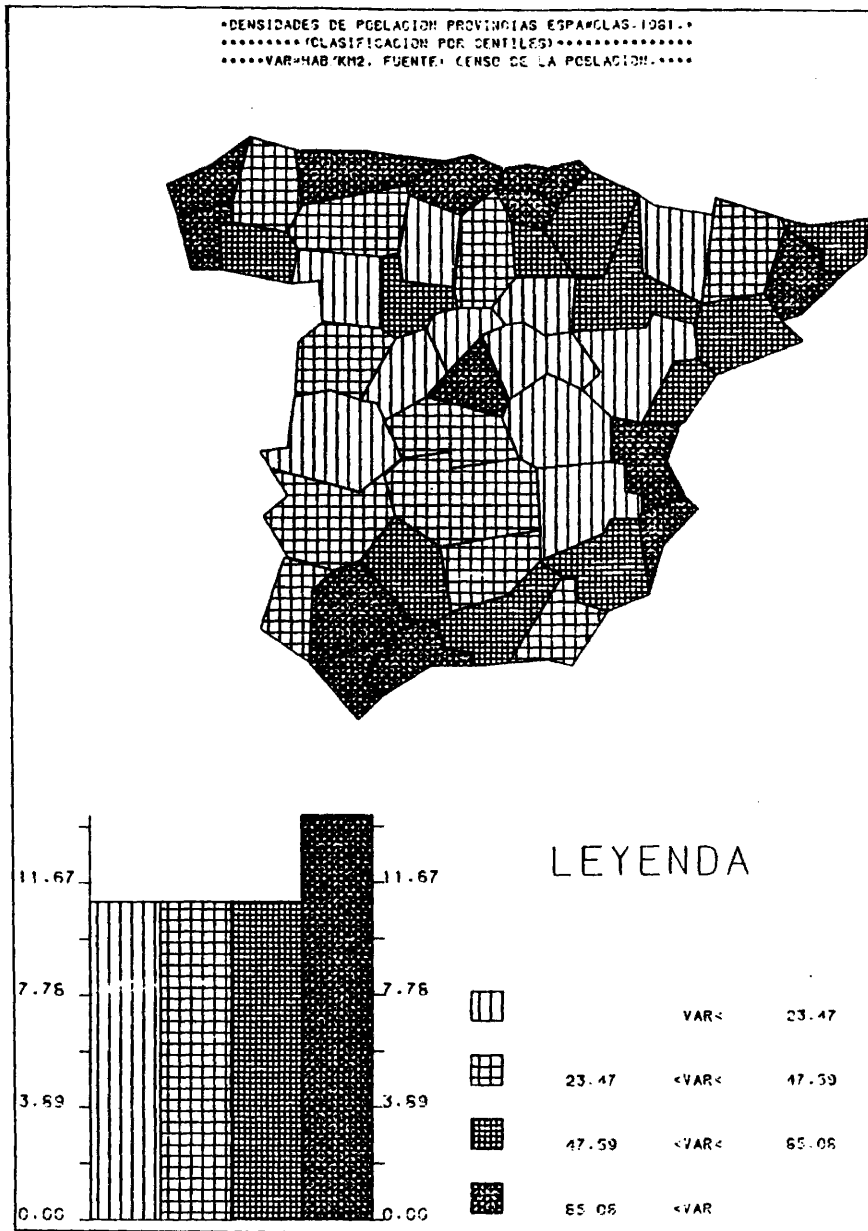


Fig.2.17.

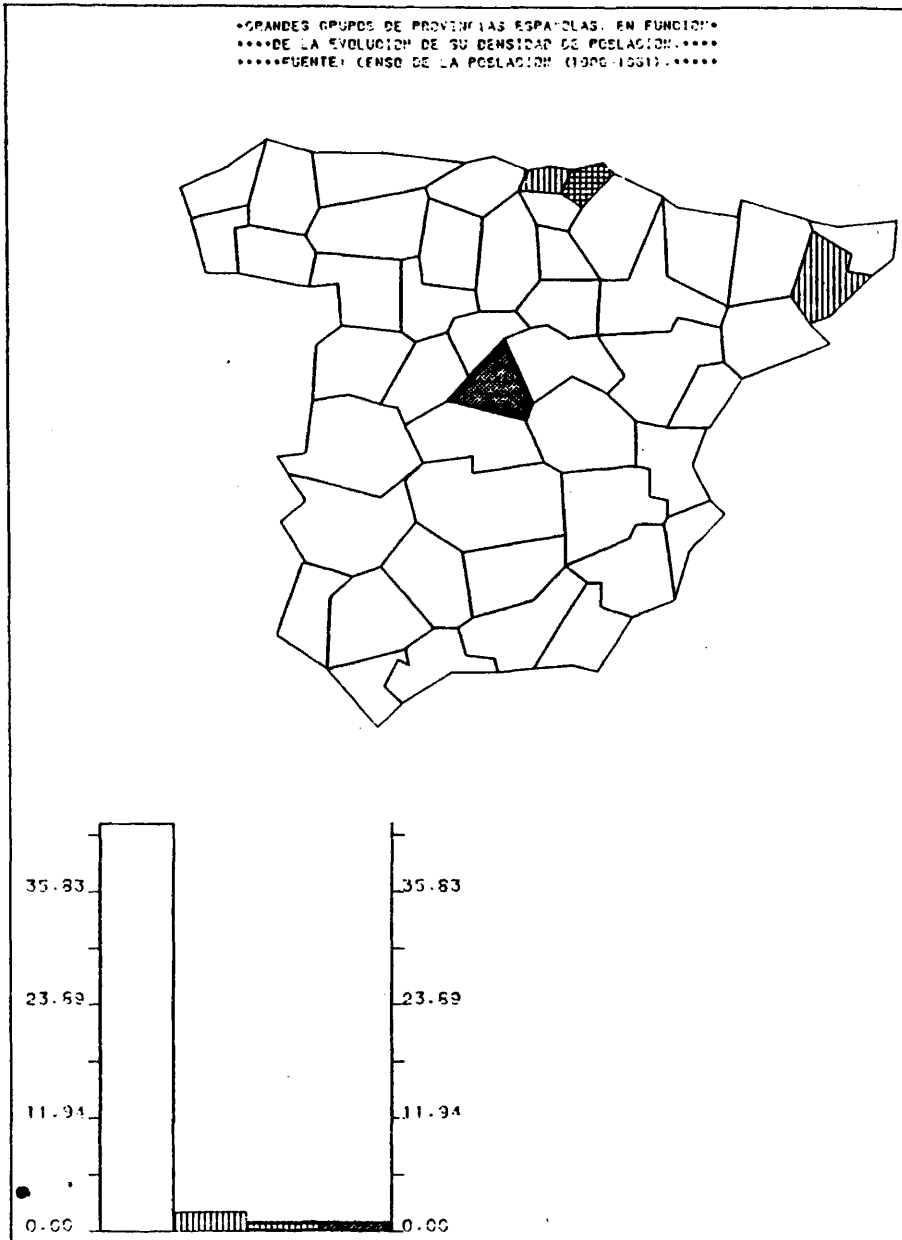


Fig.2.18.



NOTAS AL CAPITULO 2.

(1) En estos últimos meses se han desarrollado algunos programas de corrección semiautomática de pruebas de digitización, pero no los presento en este momento por encontrarse todavía en experimentación.

(2) La redacción original de esta subrutina se debe a Miguel García Ferrández. Mi aportación al texto actual ha consistido en la solución de los problemas de límite y colinealidad, en la redacción de la subrutina REORG -que interviene en la resolución de dichos problemas- y en la documentación del texto definitivo.

(3) Esta solución al problema de tramados simples de inclinación variable se debe a Miguel García Ferrández.

CAPITULO 3. CARTOGRAFIA TEMATICA EN TRES DIMENSIONES: BLOQUES  
DIAGRAMA EN PERSPECTIVA.

---

	<u>Págs.</u>
3.1. INTRODUCCION .....	160
3.2. ESPACIO TRIDIMENSIONAL Y PROYECCIONES ISOMETRICAS...	161
3.3. RAYADO DE POLIGONOS EN TRES DIMENSIONES .....	168
3.4. MODELO DE REPRESENTACION DE VOLUMENES EN PERSPECTIVA ISOMETRICA. EL PROBLEMA DE LA OCULTACION DE LINEAS..	174
3.5. DIVERSAS ALTERNATIVAS DE CONSTRUCCION DE UN VOLUMEN TE MATICO .....	177
3.5.1. El programa DIABLOCK. ....	177
3.5.1.1. Tipo de interpolación que contempla y datos que requiere coherentemente, para su funcionamiento .....	178
3.5.1.2. Listado del programa y requisitos de uso .....	181
3.5.1.3. Algunos ejemplos .....	190
3.5.1.4. Localización de contornos en un blo- que diagrama. El programa MOSAICO3..	190
3.5.2. El programa TREND .....	198
3.5.2.1. Tipo de interpolación que contempla y datos que requiere coherentemente, para su funcionamiento .....	198
3.5.2.2. Listado del programa y requisitos de uso .....	199
3.5.2.3. Algunos ejemplos .....	212
3.5.3. El programa PRISMAS. ....	212
3.5.3.1. Tipo de interpolación que contempla y datos que requiere coherentemente, para su funcionamiento .....	212
3.5.3.2. Listado del programa y requisitos de uso .....	214
3.5.3.3. Algunos ejemplos .....	226

### 3.1. INTRODUCCION.

Presentamos en este capítulo tres programas de aplicación de técnicas de representación automática de sólidos geométricos a la definición de modelos de cartografía de variables espaciales en tres dimensiones.

El fundamento básico de este tipo de representaciones lo constituye la descripción de la variación espacial de una característica por una superficie tridimensional, que hace corresponder a cada punto del plano horizontal, definido por sus coordenadas geográficas, un valor en la dimensión vertical, que es el peso de la característica en el punto en cuestión. Para definir esta superficie tridimensional es necesario recurrir a procedimientos de interpolación.

Una vez construido el volumen a representar, se procede a proyectarlo sobre el plano de dibujo, considerando -en el caso de las proyecciones isométricas, que es el que ha sido contemplado- que no existen puntos de fuga, esto es, que todas las líneas de la proyección constituyen un haz de paralelas.

Este tipo de procedimientos, reservan una componente lineal en el plano de dibujo -espacio imagen- a cada una de las componentes ortogonales del espacio origen. De esta manera, la característica a representar tiene la expresión que propicia una más exacta percepción de la magnitud del fenómeno. Además, por esta misma razón, la variación espacial de la característica puede ser tratada de una manera continua, haciendo corresponder a cada valor real una representación exactamente proporcionada -nótese, en cambio, que en el procedimiento de cartografía temática mediante coropletas se verifica una clasificación previa de las unidades espaciales en grupos homogéneos, asignando luego a cada grupo una representación común-.

Otra ventaja importante de este tipo de modelos de representación, la configura su especial adecuación para expresar la evolución espacio-temporal de una determinada característica, ya que los diversos volúmenes resultantes de la plasmación de los distintos momentos temporales, tienen todos una lectura idéntica y directamente derivable de la forma que adoptan -sin tener que recurrir en cada caso a una leyenda que explicita la relación entre la imagen plástica y el fenómeno representado-. Como veremos en el siguiente capítulo de la tesis, la animación gráfica de volúmenes tridimensionales es un importante instrumento de análisis de procesos espacio-temporales.

Las primeras aplicaciones de este tipo de técnicas a modelos de representación de variables georeferenciadas se remontan a los

primeros 70. De esa fecha es el programa BLCK, desarrollado por el Departamento de Geografía de la Universidad de Michigan. También corresponde a este primer momento de definición de modelos de representación en tres dimensiones el programa SYMVU, desarrollado por el "Harvard Laboratory for Computer Graphics and Spatial Analysis", que actualmente ya no se encuentra disponible, por haber sido sustituido por otro programa más reciente -ASPEX-.

Otros procedimientos que causan este mismo tipo de resultados gráficos son: ISOMET, desarrollado por el Departamento de Geografía de la "Northwestern University", en 1971; SURGE desarrollado por el Departamento de Geografía de la Universidad de "Rhode Island", en 1971; PERS, desarrollado por el Centro de Cálculo de la Universidad de "British Columbia", en 1972; MAP3D, desarrollado por el Departamento de Geografía de la Universidad de Minnesota y TRID, desarrollado por el "Computer Institute for Social Science Research. Michigan State University", en 1972.

Más recientes son los programas desarrollados por el "Harvard Laboratory for Computer Graphics and Spatial Analysis" a partir de 1975. El programa ASPEX, disponible a partir de 1976, establece bloques diagrama de variables georeferenciadas en perspectiva oblicua, utilizando diversos periféricos gráficos -plotter de pluma, plotter electrostático y pantalla de rayos catódicos-. En el sistema ODYSSEY, concluido en 1980, existen también procedimientos de elaboración automática de bloques diagrama, aunque el modelo de representación, por aristas, es distinto al que hemos considerado en nuestro trabajo.

Por lo que respecta a las publicaciones, resultan de especial interés sobre el tema los trabajos de Douglas /49/, Sprunt /157 y 158/, Newman y Sproull /125/, Mallet /101/, Hackathorn /71/ y Brodlie /28/, en cuanto al establecimiento de modelos de representación de volúmenes en perspectiva, y los de Crawford y Marks /43/, Rowles /146/ y Monmonier /119/, en cuanto a los problemas de percepción y optimización en este sentido de los procedimientos de representación en tres dimensiones.

### 3.2. ESPACIO TRIDIMENSIONAL Y PROYECCIONES ISOMETRICAS.

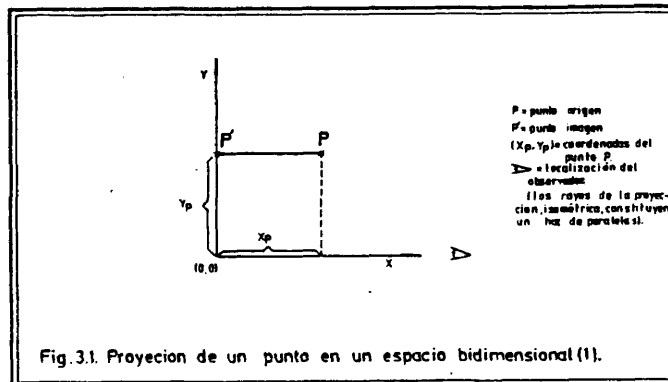
El primer problema que es necesario resolver, en el conjunto de tareas que están implicadas en la obtención automática de una vista en perspectiva de un volumen tridimensional, es el cálculo de las coordenadas de un punto del sólido a representar -proyectar- en el plano de dibujo -espacio bidimensional-, en función de las coordenadas de este punto en el espacio real de tres dimensiones.

Una vez solucionado este problema, se puede diseñar un procedimiento iterativo que calcule las coordenadas de las imágenes de los puntos significativos del sólido y las una mediante segmen

tos rectilíneos, obteniendo así una imagen aproximada del volumen a representar. Esta solución del problema es la que se ha adoptado en todos los procedimientos que se presentan en este capítulo.

Toda proyección de un punto significa la anulación de una de sus coordenadas, la que coincide con la dirección de la visual desde el observador a dicho punto.

Supongamos un caso sencillo de proyección de un punto en un espacio bidimensional (Fig. 3.1.).



La proyección del punto P sobre el eje de las Y, P', tiene por coordenadas  $(0, Y_p)$ , o lo que es lo mismo, al proyectar el punto P, se ha anulado su coordenada X, que es la que coincide con la dirección de la visual, en el caso de la figura que antecede, desde el observador al punto en cuestión.

Vamos ahora a considerar un caso en que la dirección de la visual del observador al punto no coincida con ninguno de los ejes de referencia del punto a proyectar (Fig. 3.2.).

Si giramos el sistema de referencias un ángulo  $\alpha$  hasta que coincida el eje de la X con la dirección de la nueva visual, las coordenadas del nuevo punto imagen P" serán  $(0, Y_{p''})$ . Conocido el ángulo  $\alpha$ , que identifica la posición del observador, y las coordenadas del punto a proyectar en el primitivo sistema de referencias  $(X_p, Y_p)$ , podemos calcular las coordenadas del nuevo punto imagen en el sistema de referencias girado (Fig. 3.3.).

En la figura 3.3., que coincide con la situación descrita en la figura inmediatamente anterior, se cumple la siguiente rela

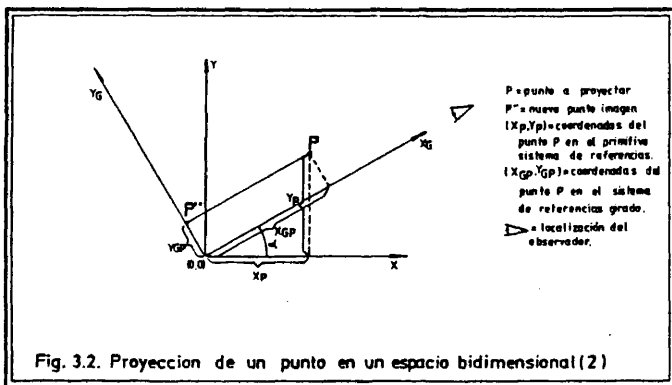


Fig. 3.2. Proyeccion de un punto en un espacio bidimensional(2)

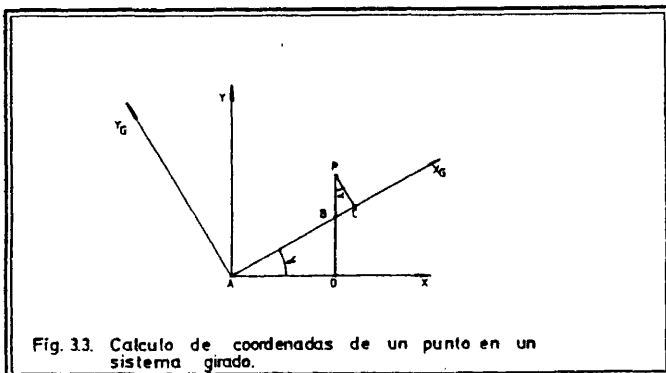


Fig. 3.3. Calculo de coordenadas de un punto en un sistema girado.

ción /3.1/  $PC/PB = \cos \alpha$ , por ser el triángulo  $\triangle PBC$  rectángulo en C.

Por otra parte, al ser el triángulo  $\triangle ABD$  rectángulo en D, se cumple la relación /3.2/  $BD/AD = \operatorname{tg} \alpha$ .

La expresión /3.1/ puede escribirse de la siguiente manera:

$$/3.3/ \quad PC = PB \cdot \cos \alpha \quad .$$

A su vez

$$/3.4/ \quad PB = PD - BD \quad .$$

Despejando BD en /3.2/, tenemos que

$$/3.5/ \quad BD = \operatorname{tg} \alpha \cdot AD \quad .$$

Ahora bien, si prestamos atención a las figuras 3.2 y 3.3 simultáneamente, podemos comprobar las siguientes equivalencias:

$$PC = Y_{GP}, \quad PD = Y_p \quad \text{y} \quad AD = X_p \quad .$$

Podemos, por tanto, reescribir las ecuaciones /3.3/, /3.4/ y /3.5/:

$$/3.6/ \quad Y_{GP} = PB \cdot \cos \alpha$$

$$/3.7/ \quad PB = Y_p - BD$$

$$/3.8/ \quad BD = \operatorname{tg} \alpha \cdot X_p$$

Sustituyendo en /3.7/ BD, por su valor en /3.8/, resulta

$$/3.9/ \quad PB = Y_p - X_p \cdot \operatorname{tg} \alpha$$

Sustituyendo en /3.6/ PB, por su valor en /3.9/, obtenemos la siguiente expresión:

$$/3.10/ \quad Y_{GP} = (Y_p - X_p \cdot \operatorname{tg} \alpha) \cdot \cos \alpha = Y_p \cdot \cos \alpha -$$

$$- X_p \cdot \frac{\operatorname{sen} \alpha}{\cos \alpha} \cdot \cos \alpha = Y_p \cdot \cos \alpha - X_p \cdot \operatorname{sen} \alpha$$

Vemos, por tanto, que, conocido el ángulo que identifica la posición del observador y las coordenadas de cualquier punto en el sistema de referencias original, los valores de las coordenadas de su imagen, son directamente deducibles. Ello permite automatizar estos cálculos.

Hasta ahora hemos considerado el caso más sencillo, en el que se contempla la proyección de un punto del plano -espacio bi dimensional- sobre una línea -espacio unidimensional-.

De igual manera se procede en todos los casos en que se trata de proyectar un punto de un espacio  $n$ -dimensional sobre un espacio  $n-1$ -dimensional: se gira el sistema de referencias original, hasta que coincida uno de los ejes con la dirección de la visual desde el observador al punto, ignorando a partir de ese momento la coordenada del punto sobre dicho eje y considerando las  $n-1$  restantes en el sistema girado.

En la obtención de vistas de volúmenes en perspectiva, el espacio origen es siempre tridimensional y la proyección se efectúa sobre un espacio bidimensional -plano de dibujo- (Fig. 3.4.).

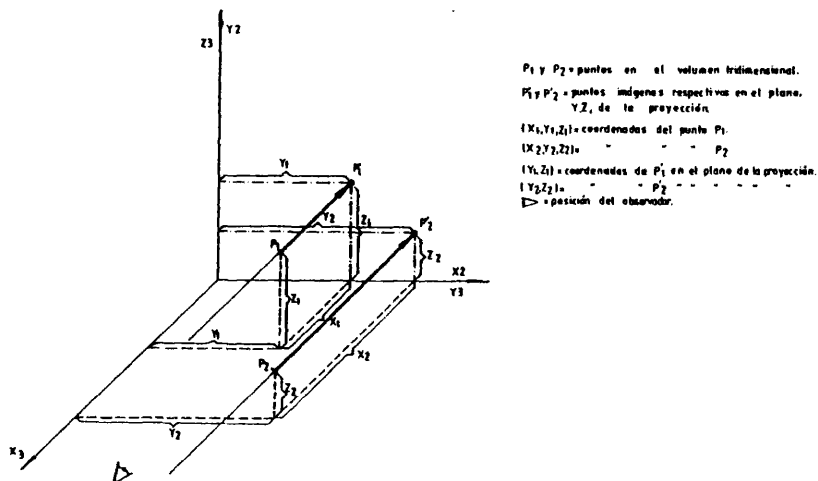


Fig 3.4. Proyecciones en el plano (Y,Z) de puntos del espacio tridimensional (X,Y,Z).

Consideraremos la situación más sencilla que se puede producir. La visual desde el observador a los puntos a proyectar coincide con la de su coordenada  $X$  en el sistema de referencias tridimensional. De forma análoga al caso descrito en la figura 3.1., las imágenes de  $P_1$  y  $P_2$  son respectivamente  $P'_1$  y  $P'_2$ , que son los puntos situados en el plano  $(Y,Z)$ , cuyas coordenadas son respectivamente las  $Y, Z$  de  $P_1$  y  $P_2$ , después de prescindir de la coordenada  $X$  de éstos.

En los casos en que la dirección de la visual no coincide con la del eje  $X$  de referencias originales, se procede, coherentemente con lo descrito en la figura 3.2., a girar el sistema de re

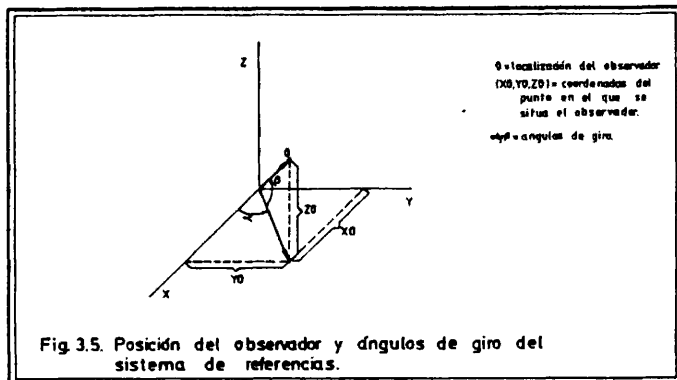


ferencias hasta que coincida la dirección del eje X con la de la visual, mediante dos giros sucesivos, uno en torno al eje Z y otro en torno al eje Y girado.

Los valores de las nuevas coordenadas ( $X_G, Y_G, Z_G$ ) de cualquier punto en el sistema girado, se calculan en base a sus coordenadas ( $X, Y, Z$ ) en el sistema de referencias original, y a los valores de los dos ángulos de giro necesarios para llevar el eje X sobre la visual desde el observador al punto.

Finalmente se asume que las coordenadas del punto imagen sobre el plano de la proyección son respectivamente  $Y_G$  y  $Z_G$ .

Se ha diseñado una subrutina, VISTA (1), que calcula automáticamente la matriz de transformación que se utiliza para calcular posteriormente las coordenadas en el espacio de dibujo de cualquier punto, en función de sus coordenadas originales y de la posición del observador -que ha de ser descrita por sus coordenadas X, Y, Z-. Esta subrutina asume que la proyección se realiza siempre sobre el plano (Y,Z) (Fig. 3.5.).



En la figura precedente, para llevar el eje X del sistema de referencias sobre la localización del observador, son necesarios dos giros sucesivos,  $\alpha$  y  $\beta$ .

La subrutina VISTA parte de las coordenadas  $X_0, Y_0, Z_0$ , del punto de vista y calcula el valor de los ángulos  $\alpha$  y  $\beta$ :

$$/3.11/ \alpha = \text{arctg } Y_0/X_0$$

$$/3.12/ \beta = \text{arctg } Z_0/(\sqrt{X_0^2 + Y_0^2})$$

A continuación se evalúan las funciones trigonométricas de  $\alpha$  y  $\beta$  necesarias para el cálculo de coordenadas en el sistema girado:  $\text{sen } \alpha$ ,  $\text{cos } \alpha$ ,  $\text{sen } \beta$  y  $\text{cos } \beta$ , y se almacenan matricialmente. El cálculo de la transformación compuesta de los dos giros se deriva del producto de las matrices donde han sido almacenadas las funciones trigonométricas de  $\alpha$  y de  $\beta$ . La matriz  $\bar{A}$  (3,3) resultante, es tal que:

$$/3.13/ \quad /X_G, Y_G, Z_G/ = /X, Y, Z/ \cdot \begin{array}{c} A \\ \left| \begin{array}{ccc} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{array} \right| \end{array}$$

$(X_G, Y_G, Z_G)$  = coordenadas del punto a proyectar en el sistema girado.  
 $(X, Y, Z)$  = coordenadas del punto a proyectar en el sistema original.

La subrutina TRN32 (2) calcula las coordenadas en el sistema girado de cualquier punto, en función de las coordenadas de éste en el espacio origen y de la matriz A. Ignora la coordenada  $\bar{X}$  del punto en el sistema girado y asume que las coordenadas X, Y del punto imagen en el plano de la proyección -dibujo- son respectivamente, las coordenadas Y, Z en el sistema girado.

Los textos de las subrutinas VISTA y TRN32, así como el de la subrutina PLOT3, que efectúa los mismos cálculos que TRN32 y, además, causa el movimiento de la pluma de un plotter al punto cuyas coordenadas se han calculado previamente, se encuentran en el apéndice final.

Antes de finalizar el presente apartado, vamos a expresar en términos algorítmicos las operaciones descritas hasta el momento, con vistas a describirlas con mayor precisión.

Sean  $X_0, Y_0, Z_0$ , las coordenadas del punto de vista; P, la lista de las coordenadas de los puntos significativos del volumen a representar,  $P = \{(X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), \dots, (X_m, Y_m, Z_m)\}$ ; A, la matriz de transformación que calcula VISTA y que utiliza TRN32. Sea PLOT (X, Y, IP) una instrucción que lleva la pluma levantada sin dibujar al punto (X, Y) si IP=3 y bajada -dibujando- si IP=2.

El algoritmo que causa el que se reproduzca una vista en perspectiva del volumen deseado, adopta la siguiente estructura.

TRID (P, X0, Y0, Z0)

TRID1. Lee X0, Y0, Z0.

TRID2. VISTA (X0, Y0, Z0).

TRID3. I = 0

TRID4. I = I+1

- TRID5. Lee el elemento I de la lista  $P = \{(X_1, Y_1, Z_1) \dots (X_m, Y_m, Z_m)\}$
- TRID6. TRN32 (P(I), XP, YP)
- TRID7. Si I=1, IP=3; si no, IP=2
- TRID8. PLOT (XP, YP, IP)
- TRID9. Si se ha leído el último elemento de P. TERMINA, si no, vuelve a TRID4.

### 3.3. RAYADO DE POLIGONOS EN TRES DIMENSIONES.

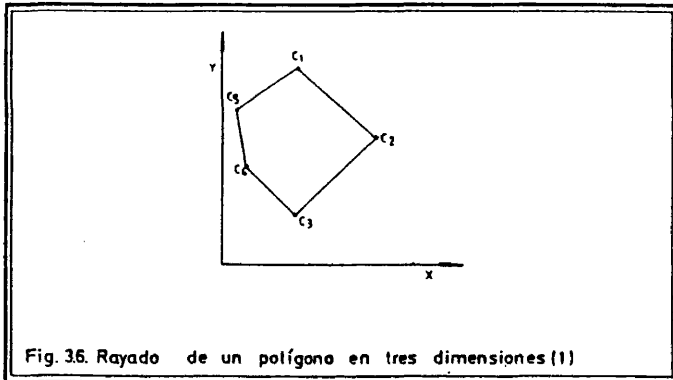
Hasta ahora sólo hemos considerado los problemas de cálculo de coordenadas en el espacio imagen de puntos de un volumen tridimensional, suponiendo conocidos los puntos significativos de dicho volumen, de cuya unión mediante segmentos rectilíneos se deriva su representación.

En este apartado nos planteamos, precisamente, un modelo de elaboración automática de la lista de triplas de coordenadas que describe adecuadamente un volumen que es necesario construir, a partir de una determinada información temática y espacial. Atendiendo a la notación que hemos utilizado al final del apartado anterior, se trata de derivar la lista P -que normalmente no es nada un dato de partida- de la información que se solicita al usuario para la elaboración de un bloque diagrama temático.

Existen muy diversos métodos para lograr este objetivo. En esta tesis se ha recurrido al procedimiento de rayar en tres dimensiones los polígonos de la referencia espacial que aporta el usuario, en función de los valores, que también introduce, de una determinada característica que afecta a dichos polígonos. La razón de ello ha sido el disponer de un software de rayado de polígonos en dos dimensiones muy depurado, diseñado previamente para resolver otro tipo de problemas planteados en el segundo capítulo de esta tesis. La subrutina RAY, ya descrita, es la herramienta básica en este tipo de procedimientos.

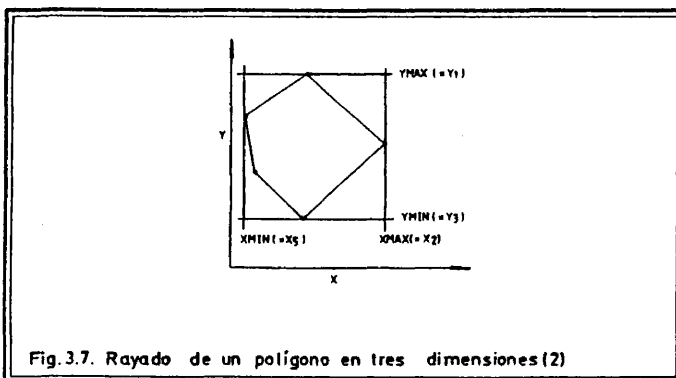
Junto a los procedimientos de rayado de polígonos, ya definidos, ha sido necesario establecer diversas rutinas de interpolación, para resolver adecuadamente problemas que son planteados en este capítulo.

Supongamos que se pretende obtener una vista en perspectiva de un volumen que exprese el valor de una característica en el interior de un polígono -en nuestro caso, un contorno geográficamente significativo-, que ha sido digitizado previamente por el usuario (Fig. 3.6.).

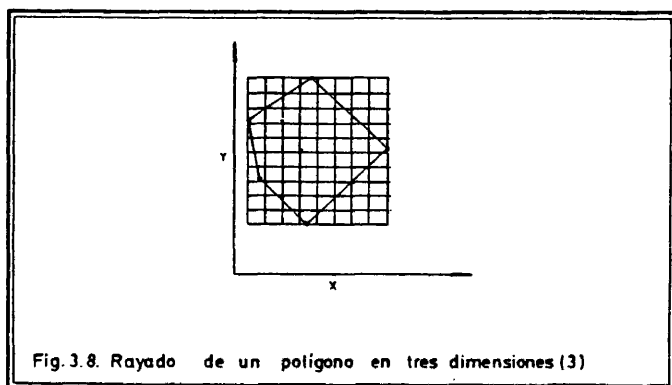


Sea  $C$ , la lista de coordenadas de los puntos que definen el polígono  $C = \{(X_1, Y_1), (X_2, Y_2), \dots, (X_5, Y_5)\}$  y  $V$  el valor de la característica en el interior del polígono.

Recorriendo la lista  $C$  podemos descubrir los valores máximos y mínimos en  $X$  e  $Y$ , lo que equivale a conocer las coordenadas del rectángulo que circunscribe el polígono (Fig. 3.7.).



Posteriormente, podemos dividir los intervalos /YMIN, YMAX/ y /XMIN, XMAX/ en un número de segmentos de igual magnitud, estableciendo una retícula ortogonal (Fig. 3.8.).



Conocidos los vértices del rectángulo -como así es- y los parámetros de la retícula definida (valor en X y en Y), que son función de aquéllos y del número de divisiones, se conocen los valores de las coordenadas X,Y de todos los nodos de la malla.

Tenemos, por tanto, ya definidas dos de las tres coordenadas de los puntos que nos van a servir para representar el volumen. Vamos a ver ahora cómo se calcula la coordenada Z de cada uno de estos puntos, en función del peso de la característica en el interior del polígono. Para ello asumimos que el valor de la coordenada Z de los puntos exteriores al polígono es cero y que el valor de ésta en los puntos interiores es V o un valor numérico proporcional a V.

El problema se reduce, por tanto, a conocer qué nodos de la retícula son interiores al polígono y cuáles exteriores. Esto se resuelve recurriendo a la subrutina RAY.

Por sucesivas llamadas a RAY, se pueden calcular todos los puntos de corte de las rayas horizontales con el polígono descrito. Vamos, pues, a fijarnos en el caso particular de una de ellas, para describir cómo se resuelve (Fig. 3.9.).

Sea  $\Delta_y$  la separación entre rayas horizontales de la retícula y  $\Delta_x$  la separación entre rayas verticales.

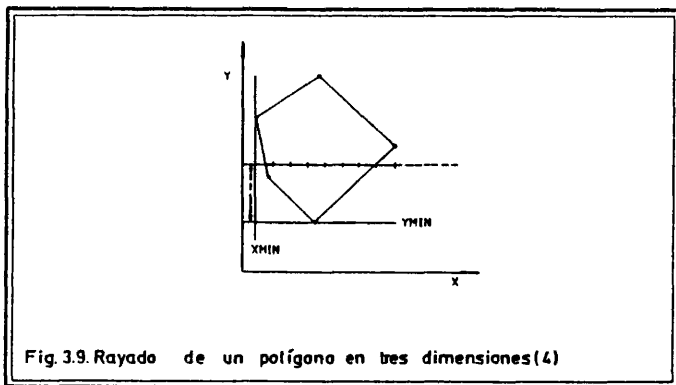


Fig.3.9. Rayado de un polígono en tres dimensiones(4)

Conocemos la coordenada Y de todos los puntos de la retícula que se encuentran sobre la raya dibujada en la figura 3.9.:  $Y = Y_{MIN} + \Delta y \cdot n$ ;  $n = n^\circ$  de orden de la línea en la retícula. También conocemos la coordenada X de cada uno:  $X = (X_{MIN} + \Delta x \cdot (i-1))$ ,  $i=1,9$ .

La llamada a la subrutina RAY (C,  $(Y_{MIN} + \Delta y \cdot n)$ , XC) (3), nos devuelve en XC las coordenadas X de los puntos de corte de la recta  $Y = Y_{MIN} + \Delta y \cdot n$  con el polígono descrito por la lista C.

Se puede, por tanto, conocer qué nodos de la malla en cada recta son interiores al polígono y cuáles exteriores. Serán interiores aquéllos en que se cumpla la relación:

$X$  = coordenada X del punto.

/3.14/  $XC_I \leq X < XC_P$  .  $XC_I$  = coordenada X de un punto de corte impar.

$XC_P$  = coordenada X de un punto de corte par.

Serán exteriores aquéllos en que se cumpla la relación:

/3.15/  $XC_P \leq X < XC_I$

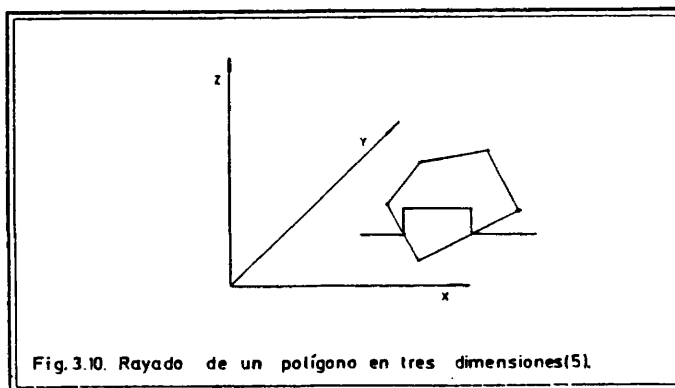
Por este procedimiento hemos podido definir la tercera coordenada (Z) de todos los puntos significativos para la construcción del volumen y podemos, por tanto, proceder a pintar sus proyecciones sobre el plano de dibujo recurriendo a VISTA a TRN32 y a PLOT,

como vimos en el apartado anterior.

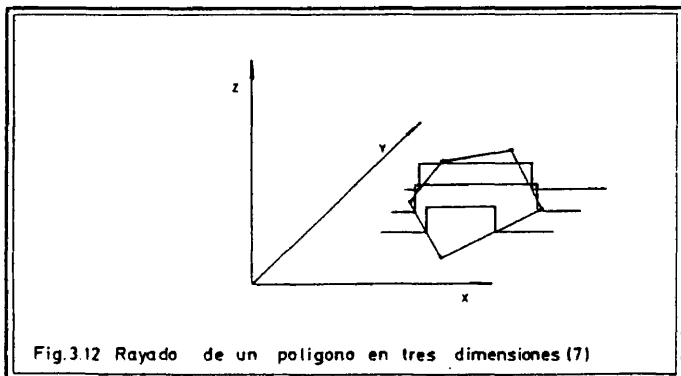
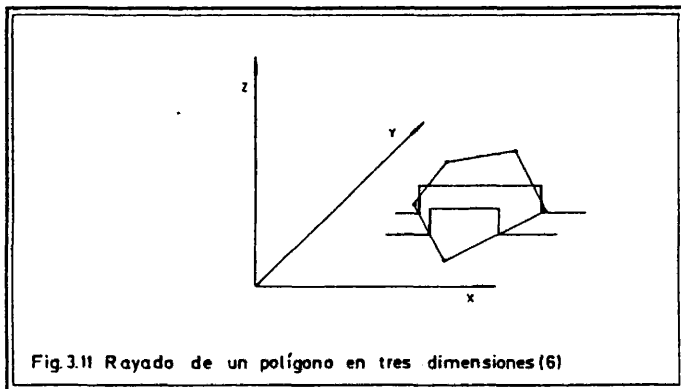
Más concretamente, el procedimiento que hemos diseñado, en vez de calcular previamente las coordenadas  $(X,Y,Z)$  de todos los puntos de la retícula y unir posteriormente con segmentos rectilíneos todas sus imágenes en el plano de dibujo, considera solamente, antes de dibujarlos, los puntos que se encuentran en una línea horizontal de la retícula. El dibujo del volumen cuya proyección isométrica se pretende representar se realiza, por tanto, línea a línea. Es por esto por lo que nos referimos a esta tarea en términos de rayado de polígonos -que son las bases de los volúmenes a representar- en tres dimensiones.

Hablando con propiedad, la línea que se dibuja en cada iteración constituye la intersección del volumen a representar con un plano vertical. Podemos decir, por tanto, que el volumen se presenta por las intersecciones de éste con una serie de planos verticales que se alejan uniformemente del observador.

En las figuras 3.10., 3.11, y 3.12. se describe gráficamente el resultado de las tres primeras iteraciones del programa de dibujo de bloques diagrama en perspectiva, utilizando como ejemplo el polígono a que nos estamos refiriendo en este apartado.



Para describir el funcionamiento básico de los programas de este capítulo, hemos considerado el caso más sencillo, que contempla la obtención de una vista en perspectiva de un volumen temático en función de un único contorno, al que afecta un determinado peso, de una característica, que se considera uniformemente distribuido en el interior de la zona afectada.





Ahora bien, las situaciones que normalmente se contemplan son más complejas. Así, de los tres programas que presentamos, dos de ellos -DIABLOCK y TREND- construyen un bloque diagrama sobre un polígono, en función del peso de la característica en diversos puntos muestrales interiores y de las coordenadas de localización de estos puntos en el plano (X,Y) horizontal. Por otra parte, el programa PRISMAS causa un volumen en perspectiva que es el resultado de la consideración del peso de una característica sobre diversos contornos geográficamente significativos.

Para poder resolver adecuadamente estas situaciones complejas, ha sido necesario recurrir a procedimientos de interpolación que se explican más adelante, en los apartados dedicados a cada uno de los programas citados.

#### 3.4. MODELO DE REPRESENTACION DE VOLUMENES EN PERSPECTIVA ISOMETRICA. EL PROBLEMA DE LA OCULTACION DE LINEAS.

Para obtener una solución satisfactoria a la representación de volúmenes mediante rayados de polígonos en tres dimensiones, es necesario, todavía, tener en cuenta una última cuestión, que supone la eliminación automática de las líneas que son ocultadas a la vista por otras líneas previamente dibujadas.

En todo cuerpo opaco, las porciones en resalte del mismo más próximas al observador ocultan siempre las zonas deprimidas que inmediatamente las suceden. Por ello es necesario ignorar en el dibujo del bloque diagrama las partes no visibles de cada línea de intersección del volumen con cada uno de los planos verticales que se siguen en profundidad. La subrutina CACHE, que resuelve el problema, es una adaptación del modelo descrito por Mallet /101/, cap. II-4-3, pp. 47 y ss.

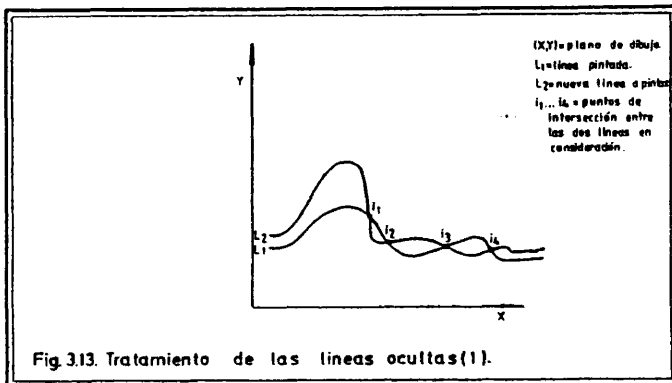
El procedimiento de ocultación de líneas que se explica a continuación es uno entre los muchos posibles, y el más apropiado al tipo de descripción de volúmenes, por sucesión de intersecciones con planos verticales, que se ha adoptado. El modelo no es aplicable a otros supuestos de representación de sólidos geométricos -por ejemplo, a los de representación por aristas-, ya que cada sistema de descripción de volúmenes requiere un tipo acorde de tratamiento de líneas ocultas.

Vamos a describir brevemente el procedimiento de ocultación de líneas que se ha adoptado.

En primer lugar es necesario precisar, de acuerdo con lo descrito en las figuras 3.10., 3.11. y 3.12., que el dibujo del diagrama se realiza iterativamente línea a línea, y que antes de producir una nueva línea, con excepción de la primera de ellas de la que se tiene seguridad de que no es ocultada por ninguna ante

rior-, se recurre al procedimiento a que nos estamos refiriendo para conocer sus partes visibles, que son las únicas que serán dibujadas.

La subrutina CACHE, que utiliza, a su vez, las subrutinas MERGE, INTER e INTP para la realización de determinadas funciones, ignora el significado de las líneas que compara y trabaja sólomente con coordenadas de puntos en el plano de dibujo.

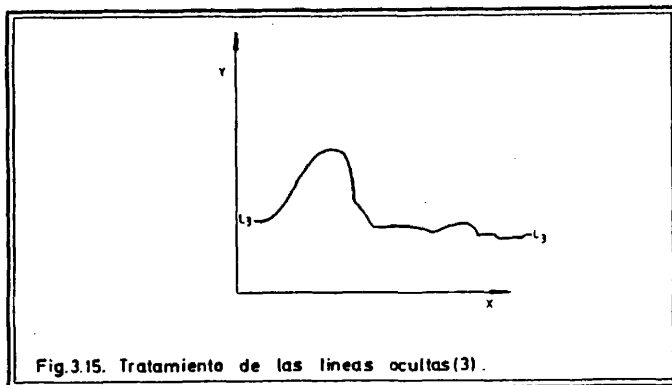
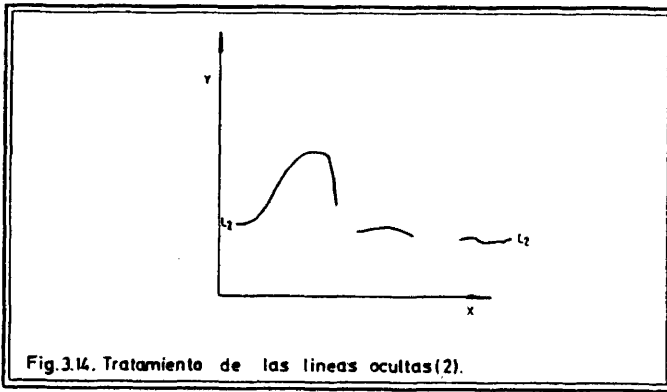


Supongamos una situación simple como la que describe la figura 3.13., en la que ha sido ya pintada la línea  $L_1$  y se desea pintar sólomente las partes visibles de la nueva línea  $L_2$ , conociendo ambas líneas por listas de pares de coordenadas que definen un conjunto discreto de puntos que las aproxima.

El resultado del funcionamiento de la subrutina CACHE, cuyo detalle puede estudiarse en el texto del apéndice final, es el siguiente.

En primer lugar se recorre la lista que describe la nueva línea, pintando sólomente las secciones que no son ocultas por la anterior (Fig. 3.14.).

En segundo término, se funden las coordenadas de los puntos de  $L_1$  y  $L_2$ , junto con las de los puntos de intersección de ambas, que describen la línea envolvente que discurre por todas las cotas máximas del nuevo dibujo, dando lugar a una nueva lista,  $L_3$ , de pares de coordenadas (Fig. 3.15.).



Finalmente se almacena esta nueva línea, así formada, para su comparación con la siguiente en el dibujo del bloque diagrama.

La subrutina CACHE, junto a otros que no son relevantes para la comprensión del procedimiento al nivel que está siendo examinado, tiene dos parámetros fundamentales de entrada y salida de información: la nueva línea a pintar y la envolvente de todas las líneas dibujadas anteriormente.

La inclusión de este procedimiento en la interacción general que causa el dibujo del bloque diagrama es de la siguiente forma.

Sea una lista LINEA =  $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$ , que describe la proyección sobre el plano de dibujo de una de las intersecciones del volumen a representar con uno de los planos verticales que se consideran, y ENVOL otra lista  $\{(X'_1, Y'_1), (X'_2, Y'_2), \dots, (X'_n, Y'_n)\}$  que describe la envolvente de todas las líneas pintadas hasta ese momento -que se corresponden con intersecciones del volumen con planos más próximos al observador-. El algoritmo H:

- H1. Construye una nueva lista LINEA =  $\{(X_1, Y_1), (X_2, Y_2), \dots, (X_n, Y_n)\}$ .
- H2. CACHE (ENVOL, LINEA).
- H3. Si el volumen ya ha sido descrito, TERMINA, si no, vuelve a H1.  
Causará el dibujo de todas las secciones visibles del bloque diagrama.

En cada una de las iteraciones del algoritmo anterior, la subrutina CACHE recibirá las descripciones de la envolvente hasta el momento (ENVOL) y de la nueva línea a pintar (LINEA), dibujará las secciones visibles de LINEA y devolverá al programa principal la descripción de la nueva envolvente (ENVOL) para que ésta sea utilizada en sucesivas iteraciones.

### 3.5. DIVERSAS ALTERNATIVAS DE CONSTRUCCION DE UN VOLUMEN TEMATICO.

#### 3.5.1. El programa DIABLOCK.

Este programa ha sido diseñado para la obtención de bloques diagrama en perspectiva que expresan la variación continua de una determinada característica en el interior de un único con torno geográficamente significativo, a partir de los valores de aquella en puntos interiores a éste.

### 3.5.1.1. Tipo de interpolación que contempla y datos que requiere, coherentemente, para su funcionamiento.

En el apartado 3.3, de este capítulo, consideramos el caso del rayado en tres dimensiones de un único polígono en función del valor de una característica uniformemente distribuida en su interior. Vimos, en aquella ocasión que el cálculo del valor de la coordenada  $Z$  de todos los puntos de la retícula ortogonal necesarios para el dibujo del volumen, se limitaba al conocimiento de si se encontraban en el exterior del polígono ( $Z=0$ ) o en su interior ( $Z$ =valor de la característica en el interior del mismo).

El caso que ahora nos planteamos es más complejo, ya que, manteniéndose la condición de que la coordenada  $Z$  de todos los puntos exteriores al polígono es nula, se pretende asignar a cada punto de la retícula interior al mismo un valor diferencial de  $Z$ , en función de la distribución del peso de la característica en una muestra espacial. Para ello es necesario recurrir a un procedimiento de interpolación en dos dimensiones.

Antes de explicar el procedimiento de interpolación que ha sido definido, vamos a comentar con un ejemplo sencillo el tipo de información que el programa DIABLOCK requiere para su funcionamiento. De esta manera pensamos que se facilitará su comprensión.

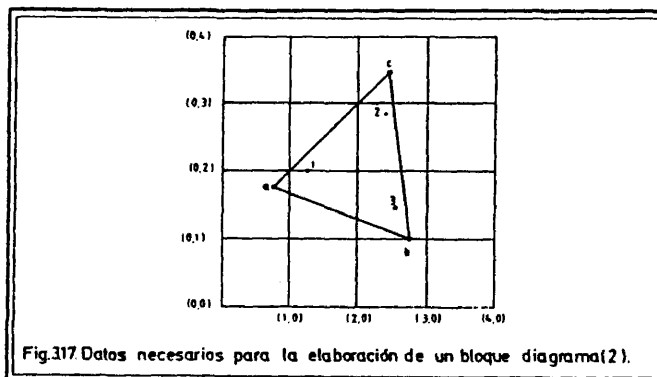
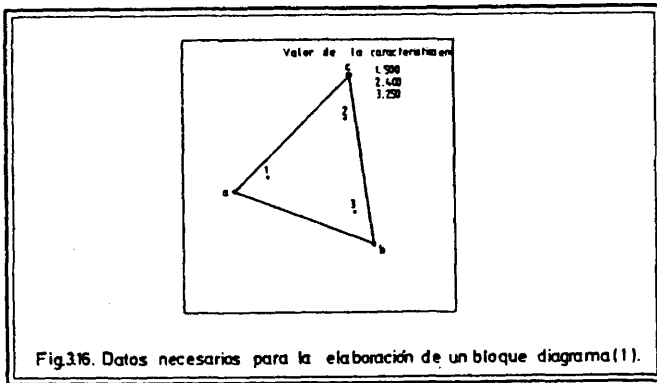
El programa necesita conocer dos conjuntos de datos fundamentales. El primero de ellos es una descripción del contorno del área en cuestión por un número discreto de pares de coordenadas que definen puntos de dicho contorno. El segundo es una lista de valores en la que se registran las dos coordenadas y el peso de la característica en cada punto de la muestra espacial.

Para preparar la información espacial que requiere el programa -coordenadas de los puntos del contorno y de la muestra-, si no se dispone de un digitizador, es necesario superponer sobre el mapa de referencia una retícula ortogonal y definir un origen de coordenadas. Posteriormente se van anotando las coordenadas de todos los puntos.

Supongamos un caso en el que se considera un contorno muy simple y en el que se ha sondeado la característica en sólo tres puntos (Fig. 3.16.).

De la figura 3.17. puede deducirse fácilmente que el contorno queda descrito por la sucesión de los tres pares de coordenadas que corresponden a los vértices del triángulo:

	X	Y
a	0.9	1.9
b	2.8	1.0
c	2.5	3.5



Por otra parte, la lista de valores de la característica en la muestra, y su localización espacial, queda definida de la siguiente manera:

	X	Y	P
1.	1.4	2.0	500
2.	2.3	2.8	400
3.	2.5	1.5	250

Pues bien, si levantamos un bloque diagrama coherente con toda esta información, la definición numérica del contorno en cuestión nos sirve para conocer qué puntos de la retícula ortogonal, que se define, son exteriores al mismo y, por tanto, qué puntos tienen una coordenada vertical (Z) nula. Esta información nos permite conocer también los puntos de la retícula interiores al polígono, e invocar para cada uno de ellos un procedimiento de interpolación que nos dará un valor de su coordenada vertical, que esté en consonancia con los valores de la característica en los puntos de la muestra espacial.

El programa DIABLOCK utiliza un procedimiento de interpolación por aproximación numérica. Todos los procedimientos de este tipo consideran que el valor de la característica en un determinado punto del mapa está condicionado por la distancia desde ese punto a los puntos muestrales más próximos, y por el valor de la característica en ellos.

Este método da lugar a superficies tridimensionales bastante complejas, y hace coincidir con bastante aproximación el valor de la superficie interpolada en un punto muestral y el valor de la característica en ese punto.

Más concretamente, la fórmula de interpolación por aproximación numérica elegida, ha sido la siguiente:

$$/3.16/ z_i = \frac{\sum_{K=1}^m (z_K / D_{iK}^2)}{\sum_{K=1}^m (1 / D_{iK}^2)}$$

Donde:

$z_i$  = valor interpolado de la característica en el punto i de la retícula.

m = número de puntos que componen la muestra espacial.

$z_K$  = valor de la característica en el punto K de la muestra.

$D_{iK}$  = distancia lineal entre el punto i de la retícula y el punto K de la muestra.

Esta formulación, comparada con la expresión básica de medias ponderadas por distancias:

$$/3.17/ \quad z_i = \frac{\sum_{k=1}^m (z_k/D_{ik})}{\sum_{k=1}^m (1/D_{ik})} ,$$

concede mayor influencia a los puntos de la muestra más próximos (Mac Dougall /97/, Cap.5.5). No obstante, existen formulaciones de interpolación por aproximación numérica, que realzan todavía más la influencia local en el cálculo de valores interpolados. Tal es el caso de la fórmula que utiliza Tobler /169/:

$$/3.18/ \quad z_i = 1/2 \left\{ z_n + \left( \frac{\sum_{k=1}^m (z_k/D_{ik}^2)}{\sum_{k=1}^m (1/D_{ik}^2)} \right) \right\} ,$$

en la que  $z_n$  es el valor de la característica en el punto muestral más próximo.

#### 3.5.1.2. Listado del programa y requisitos de uso.

Reproducimos a continuación el texto, documentado, del programa DIABLOCK, en el que aparecen claramente explicitados todos los requisitos de uso del mismo.

Los textos de todas las subrutinas que utiliza se encuentran en el apéndice final, clasificados por orden alfabético.



```

C
C *****
C * PROGRAMA DE DIBUJO DE BLOQUES DIAGRAMA EN PERSPECTIVA ISOMETRI-
C * CA, QUE EXPRESAN LA VARIACION DE UNA CARACTERISTICA EN EL INTERIOR
C * DE UN AREA, A PARTIR DE LAS COORDENADAS QUE DEFINEN SU CONTORNO Y
C * DE LOS VALORES DE LA CARACTERISTICA EN DIVERSOS PUNTOS INTERIORES
C * A ELLA, DE LOS CUALES SE NECESITA CONOCER TAMBIEN SU LOCALIZACION
C * ESPACIAL MEDIANTE COORDENADAS.
C * EL VALOR DE LA FUNCION EN UN PUNTO SE CALCULA RECURRIENDO AL
C * SIGUIENTE PROCEDIMIENTO DE INTERPOLACION:
C *
C *      M      M
C *      Z(I)=(SUMA(Z(K)/D(I,K)**2))/(SUMA(1/(D(I,K)**2)))
C *      K=1      K=1
C *
C * DONDE:
C * Z(I) = VALOR DE LA FUNCION EN UN PUNTO.
C * M = NUMERO DE CENTROIDES.
C * Z(K) = VALOR DE LA CARACTERISTICA EN EL CENTROIDE K.
C * D(I,K) = DISTANCIA LINEAL ENTRE EL PUNTO Y EL CENTROIDE K.
C *
C * *****
C * EL PROGRAMA PUEDE CONSIDERAR HASTA 300 CENTROIDES Y UN CONTORNO
C * DEFINIDO POR UN MAXIMO DE 100 PUNTOS. SI SE REBASAN ESTOS LIMITES,
C * ES NECESARIO REDIMENSIONAR LOS ARRAYS XV,YV,PV,D,B,X E Y.
C * *****
C
C NOMBRE DEL PROGRAMA: DIABLOCK.
C
C LENGUAJE: FORTRAN77
C
C AUTORES: JUAN ANTONIO CEBRIAN DE MIGUEL.
C           DEPARTAMENTO DE GEOGRAFIA HUMANA.
C           MIGUEL GARCIA FERRANDEZ.
C           DEPARTAMENTO DE ECUACIONES FUNCIONALES.
C           UNIVERSIDAD COMPLUTENSE (MADRID).
C           JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C
C MEMORIA: 204 KBYTES.
C
C SUBRUTINAS NECESARIAS: ZETA, RAY, REDRG, SETG2, G1, G2, VISTA, PLOT3
C TRN32, CACHE, INTER, MERGE, INTP, SCVERT, TITULO, MARCO, PLOTMM.
C
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN TRES FICHEROS.
C
C FICHERO FT05F001 (REUNE TODOS LOS PARAMETROS DE CONTROL).
C *****
C PRIMERA TARJETA.

```

```

C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C AAAA BBBB CCCC.CCCC
C
C A = COORDENADA X DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO
C QUE DIAGRAMA.
C B = COORDENADA Y DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO
C QUE DIAGRAMA.
C C = FACTOR DE ESCALA DEL BIBUJO.
C
C TERCERA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C DDDD EEEE FFFF GGGGG HHH.MMM III.III
C
C D = COORDENADA X DEL PUNTO DE VISTA ELEGIDO.
C E = COORDENADA Y DEL PUNTO DE VISTA ELEGIDO.
C F = COORDENADA Z DEL PUNTO DE VISTA ELEGIDO.
C G = ANGULO (EN EL SENTIDO DE LAS AGUJAS DEL RELOJ) QUE SE DESEA
C QUE FORME EL RAYADO CON LA VISUAL DIRIGIDA DESDE EL PUNTO DE
C VISTA AL ORIGEN DE COORDENADAS.
C H = VALOR Y DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGO-
C NO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.
C I = VALOR X DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGO-
C NO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.
C
C CUARTA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C JJJJ KKKK.KKKK LLLL.LLLL MMMM.MMMM NNNN.NNNN
C
C J = VALOR DE LA FUNCION FUERA DEL CONTORNO.
C K = VALOR MINIMO DE LA FUNCION DENTRO DEL CONTORNO.
C L = VALOR MAXIMO DE LA FUNCION DENTRO DEL CONTORNO.
C M = FACTOR DE CORRECCION DE LA FUNCION (*).
C N = FACTOR DE CORRECCION DE LA FUNCION (+).
C
C QUINTA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C (2F6.3,F10.2)
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS
C VALORES DE LAS COORDENADAS DE LOS CENTROIDES Y EL VALOR DE LA CARAC-
C TERISTICA EN CADA CENTROIDE.
C
C SEXTA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C (8F10.0)
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 9 LOS
C VALORES DE LAS COORDENADAS DE LOS PUNTOS QUE DEFINEN EL CONTORNO EX-

```

```

C   TERIOR DEL AREA.
C
C   SEPTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   0
C
C   D = NUMERO DE TARJETAS (MAXIMO 3) QUE FORMAN EL TITULO DEL DIBUJO.
C
C   OCTAVA TARJETA (O TARJETAS).
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   BLOQUE DIAGRAMA QUE REPRESENTA.....
C
C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   /#
C
C   FICHERO FT08F001 (NUMERO DE CENTROIDES Y VALORES DE SUS COORDENADAS
C   ***** Y DE LA CARACTERISTICA EN ELLOS).
C
C   PRIMERA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   //FT08F001 DD *
C
C   SEGUNDA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   AAA
C
C   A = NUMERO DE CENTROIDES.
C
C   TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y, PESO DE
C   LA CARACTERISTICA DE CADA CENTROIDE -POR ESTE ORDEN-, CON EL FORMA-
C   TO DESCRITO.
C
C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   /#
C
C   FICHERO FT09F001 (NUMERO DE PUNTOS QUE FORMAN EL CONTORNO Y VALORES
C   ***** DE SUS COORDENADAS).
C
C   PRIMERA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   //FT09F001 DD *
C
C   SEGUNDA TARJETA.
C   1      10      20      30      40      50      70      80
COL |.....|.....|.....|.....|.....|.....|.....|
C   AAA
C
C   A = NUMERO DE PUNTOS DEL CONTORNO.
C

```

```

C TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y DE CADA
C UNO DE LOS PUNTOS -POR ESTE ORDEN-, CON EL FORMATO DESCRITO.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL. | ..... | ..... | ..... | ..... | ..... | ..... | ..... |
C /*
C .....
C .....
C .....
C
C DIMENSION XV(300),YV(300),PV(300),D(300)
C DIMENSION A(30)
C DIMENSION B(2,100)
C DIMENSION X(100),Y(100)
C DIMENSION W(3,600),V(2,600)
C REAL LINEA(2,300),LINEAA(3,300)
C CHARACTER FMT(80)
C LOGICAL LW(600)
C COMMON/XYMM/XM2,YM2,XM1,YM1,FCT
C
C XV = VECTOR DONDE SE ALMACENAN LAS COORDENADAS X DE LOS CENTROIDES.
C YV = VECTOR DONDE SE ALMACENAN LAS COORDENADAS Y DE LOS CENTROIDES.
C PV = VECTOR DONDE SE ALMACENAN LOS PESOS DE LA VARIABLE SOBRE LOS
C CENTROIDES.
C D = VECTOR DONDE SE ALMACENAN TODAS LAS DISTANCIAS EN LINEA RECTA
C ENTRE UN PUNTO INTERIOR AL AREA EN CONSIDERACION Y CADA UNO DE
C LOS CENTROIDES (EL CALCULO DE ESTAS DISTANCIAS VIENE REQUERIDO
C POR EL TIPO DE INTERPOLACION QUE SE REALIZA).
C B = MATRIZ DONDE SE ALMACENAN LAS COORDENADAS QUE DESCRIBEN EL CON-
C TORNOC DEL AREA.
C FMT = VECTOR DE CARACTERES PARA LA DESCRIPCION DEL FORMATO DE LOS DA-
C TOS APORTADOS POR EL USUARIO.
C A,X,Y,W,V,LINEA,LINEAA Y LW = POSICIONES DE MEMORIA QUE SE UTILIZAN
C PARA EL CALCULO DE TODOS LOS VALORES
C NUMERICOS NECESARIOS PARA LA CONSTRUC-
C CION DEL DIBUJO.
C (XM2,YM2) = COORDENADAS DEL VERTICE SUPERIOR DERECHO DEL MINIMO REC-
C TANGULO QUE ENMARCA EL DIBUJO.
C (XM1,YM1) = COORDENADAS DEL VERTICE INFERIOR IZQUIERDO DEL MINIMO
C RECTANGULO QUE ENMARCA EL DIBUJO.
C .....
C
C INICIALIZACION DE VARIABLES DE CONTROL DE ESCRITURA.
C
C DATA IDBS,NDIS/0,1/
C
C INICIALIZACION DE VALORES DE LA ZONA COMUN XYMM PARA AVERIGUAR EL
C RECUADRO.
C
C XM1= 1.E74
C XM2=-1.E74
C YM1= 1.E74
C YM2=-1.E74
C .....

```

```

C
C LECTURA DE DATOS.
C *****
C
C LECTURA DE PARAMETROS.
C
C   READ(5,201) XOR,YOR,FCT
C   WRITE(6,201) XOR,YOR,FCT
201  FORMAT(5X,2F5.0,F10.4)
C   READ(5,202) VX,VY,VZ,WR,DR,DS
C   WRITE(6,202) VX,VY,VZ,WR,DR,DS
202  FORMAT(5X,3F5.0,F6.0,2F8.3)
C   READ(5,203) ZB,Z0,Z1,ZF,ZINC
C   WRITE(6,203) ZB,Z0,Z1,ZF,ZINC
203  FORMAT(5X,F5.0,3F10.4,T40,F10.4)
C
C LECTURA DE LAS COORDENADAS DE LOS CENTROIDES Y DEL PESO DE LA VA-
C RIABLE EN ELLOS.
C
C   READ(8,100) N2
C   WRITE(6,100) N2
100  FORMAT(2X,I3)
C   READ(5,107) FMT
107  FORMAT(80A1)
C   WRITE(6,108) FMT
108  FORMAT('OFORMATO DE LOS DATOS = ',80A1/)
C   READ(8,FMT) (XV(I),YV(I),PV(I),I=1,N2)
C   WRITE(6,103) (XV(I),YV(I),PV(I),I=1,N2)
103  FORMAT(2X,3F12.4)
C
C LECTURA DE LAS COORDENADAS DEL CONTORNO EXTERIOR.
C
C   READ(9,100) N
C   WRITE(6,100) N
C   READ(5,107) FMT
C   WRITE(6,104) FMT
104  FORMAT('OFORMATO DEL CONTORNO = ',80A1/)
C   READ(9,FMT) (B(1,I),B(2,I),I=1,N)
C   WRITE(6,105) N,(B(1,I),B(2,I),I=1,N)
105  FORMAT(13/(2F10.3))
C
C *****
C *****
C *****
C *****
C
C SECCION DE DIBUJO.
C *****
C
C SE ABRE EL DIBUJO.
C
C   CALL PLOTS(0,0,6)
C
C *****
C
C SE CONSIDERA EL FACTOR DE ESCALA.
C
C   CALL FACTOR(FCT)

```

```

C
C SE DESPLAZA EL ORIGEN DE COORDENADAS.
C   CALL PLOT(XDR,YDR,-3)
C
C SE CALCULA LA MATRIZ DE TRANSFORMACION DE COORDENADAS EN EL ESPACIO
C DE TRES DIMENSIONES, EN COORDENADAS EN EL ESPACIO BIDIMENSIONAL DE
C DIBUJO.
C   CALL VISTA(VX,VY,VZ)
C
C SE PINTA UN TRIEDRO EN EL ORIGEN DE COORDENADAS.
C
C   CALL PLOT3(1.,0.,0.,3)
C   CALL PLOT3(0.,0.,0.,2)
C   CALL PLOT3(0.,2.,0.,2)
C   CALL PLOT3(0.,0.,3.,3)
C   CALL PLOT3(0.,0.,0.,2)
C
C SE DEFINE EL ANGULO DE RAYADO COHERENTE CON EL PUNTO DE VISTA ELE-
C GIDO Y CON LA PETICION MANIFESTADA POR EL USUARIO.
C
C   CALL SETG2(VX,VY,WRA)
C   WRA=-WRA
C   WR=WR-90.
C   WR=WR+WRA
C
C SE GIRA EL POLIGONO, SE ENMARCA Y SE CALCULA EL NUMERO DE RAYAS NE-
C CESARIAS PARA REPRODUCIR EL BLOQUE DIAGRAMA, EN FUNCION DE LA RETI-
C CULA ORTOGONAL DEFINIDA POR EL USUARIO AL DAR VALORES A DR Y A DS.
C
C   XMIN=1.E74
C   XMAX=-1.E74
C   YMAX=-1.E74
C   YMIN=1.E74
C   DO 1 I=1,N
C     CALL G1(X(I),Y(I),B(1,I),B(2,I),WR)
C     IF(YMAX.LT.Y(I)) YMAX=Y(I)
C     IF(YMIN.GT.Y(I)) YMIN=Y(I)
C     IF(XMAX.LT.X(I)) XMAX=X(I)
C     IF(XMIN.GT.X(I)) XMIN=X(I)
C 1 CONTINUE
C   WRITE(6,106) YMIN,YMAX,XMIN,XMAX
C   WRITE(6,106) (X(I),Y(I),I=1,N)
106 FORMAT(/(10X,2F10.3))
C   M=(YMAX-YMIN)/DR+1.5
C   YR=YMIN-(DR+0.06789)
C
C *****
C   M3M=0
C   MAX=600
C *****
C
C EN ESTA ITERACION SE DIBUJA EL BLOQUE DIAGRAMA.
C
C *****
C   DO 20 J=1,M
C *****

```

```

C
C 20.1. SE RAYA EL POLIGONO EN TRES DIMENSIONES.
C
  YR=YR+DR
  CALL RAY(X,Y,N,YR,A,K)
  IF(K.LE.0) GO TO 20
  I=0
  DX=DS
  XR=XMIN-DX
  A(K+1)=999999.
  ICONT=0
2  CONTINUE
  XR=XR+DX
  IF(XR.GT.XMAX+DX) GO TO 3
  IF(XR.LE.A(I+1)) GO TO 3
  I=I+1
3  CONTINUE
  CALL G2(XN,YN,XR,YR,WR)
  Z=ZB
  IF(MOD(I,2).EQ.0) GO TO 4
C
C 20.1.1. LLAMADA A LA SUBROUTINA DE INTERPOLACION.
C
  CALL ZETA(XN,YN,P,XV,YV,PV,D,N2)
  IF(IDBS.GE.NDIS)
  *WRITE(6,999) XR,YR,XN,YN,P
999 FORMAT(60X,3F10.3)
  Z=ZF+(P-ZINC)
  Z=AMAX1(Z0,Z)
  Z=AMINI1(Z1,Z)
4  CONTINUE
C
C 20.2. SE CALCULAN LAS COORDENADAS DE LA RAYA EN EL ESPACIO IMAGEN.
C
  ICONT=ICONT+1
  LINEAA(1,ICONT)=XN
  LINEAA(2,ICONT)=YN
  LINEAA(3,ICONT)=Z
  CALL TRN32(LINEA(1,ICONT),LINEAA(1,ICONT))
  GO TO 2
5  CONTINUE
C
C 20.3. SE CALCULAN LAS SECCIONES NO OCULTADAS POR SECCIONES ANTERIO-
C RES, Y SE DIBUJAN.
C
  IER=0
  CALL CACHE(V,M3M,M33,MAX,LINEA,ICONT,W2LW,0,IER,0)
  M3M=M33
C
  IF(IER.GE.6) STOP
C
C*****
20 CONTINUE
C*****
C
C*****
C
C SE DEVUELVE EL ORIGEN DE COORDENADAS AL PUNTO EN QUE SE ENCONTRABA

```

```

C   ANTES DE REALIZAR EL DIBUJO DEL BLOQUE DIAGRAMA.
C
C   CALL PLOT(-XDR,-YDR,-3)
C
C *****
C
C   EL DIBUJO DEL DIAGRAMA TRIDIMENSIONAL ESTA COMPRENDIDO ENTRE LOS
C   VERTICES (XM1,YM1) Y (XM2,YM2).
C
C   XM1=XM1+XDR*FCT
C   YM1=YM1+YDR*FCT
C   XM2=XM2+XDR*FCT
C   YM2=YM2+YDR*FCT
C
C *****
C
C   LLAMADA A LA SUBROUTINA QUE SITUA LA ESCALA VERTICAL DE LECTURA DEL
C   DIAGRAMA TRIDIMENSIONAL A LA DERECHA DEL DIBUJO.
C
C   XSC=(XM2/FCT)+3.
C   YSC=(YM2/FCT)/2.
C   CALL SCVERT(XSC,YSC,ZB,ZO,Z1,ZF,ZINC,XTOP,YTOP)
C   XM2=AMAX1(XM2,(XTOP*FCT))
C   YM2=AMAX1(YM2,(YTOP*FCT))
C
C *****
C
C   LLAMADA A LA SUBROUTINA QUE REPRODUCE EL TITULO DEL DIAGRAMA.
C
C   CALL FACTOR(1.)
C   CALL TITULO(XM1,YM2,(XM2-XM1),4.)
C   YM2=YM2+4.
C
C *****
C
C   SE ENMARCA EL DIBUJO.
C
C   CALL PLOT(0.,0.,3)
C   XMARC=XM1-1.
C   YMARC=YM1-1.
C   CALL MARCO(XMARC,YMARC,(XM2+1.-XM1),(YM2+1.-YM1))
C
C *****
C
C   SE CIERRA EL DIBUJO.
C
C   CALL PLOT(9.,-8.,999)
C
C *****
C
C   STOP
C   END

```



### 3.5.1.3. Algunos ejemplos.

En diversos trabajos y publicaciones se pueden encontrar aplicaciones concretas de este programa de representación a tareas de cartografía temática de variables georeferenciadas (ver Bosque Maurel, Cebrián de Miguel y Bosque Sendra /19/, Bosque Sendra/23/, Bosque Maurel /20/, Capel y Urteaga /31/, Cebrián de Miguel y García Fernández /35/, García Ballesteros /60/, Herrero García y Cebrián de Miguel /77/ y Requés Velasco /133/).

Presentamos a continuación un conjunto de vistas complementarias del volumen que representa la distribución espacial de la densidad de población en la España peninsular en el año 1981, como muestra de las prestaciones del programa.

La escala gráfica que aparece a la derecha del dibujo del bloque diagrama, y que no hemos comentado en los apartados precedentes, permite una lectura aproximada, en la vertical, de éste.

El marco y los rótulos también han sido reproducidos automáticamente.

### 2.5.1.4. Localización de contornos en un bloque diagrama. El programa MOSAICO3.

Para facilitar la interpretación de este tipo de representaciones, se está desarrollando un conjunto de procedimientos complementarios de ilustración del gráfico, que está todavía en experimentación.

El programa que presentamos a continuación forma parte de ese grupo, y puede ser utilizado para el reconocimiento en el bloque diagrama de contornos geográficamente significativos interiores al único contorno que considera el programa.

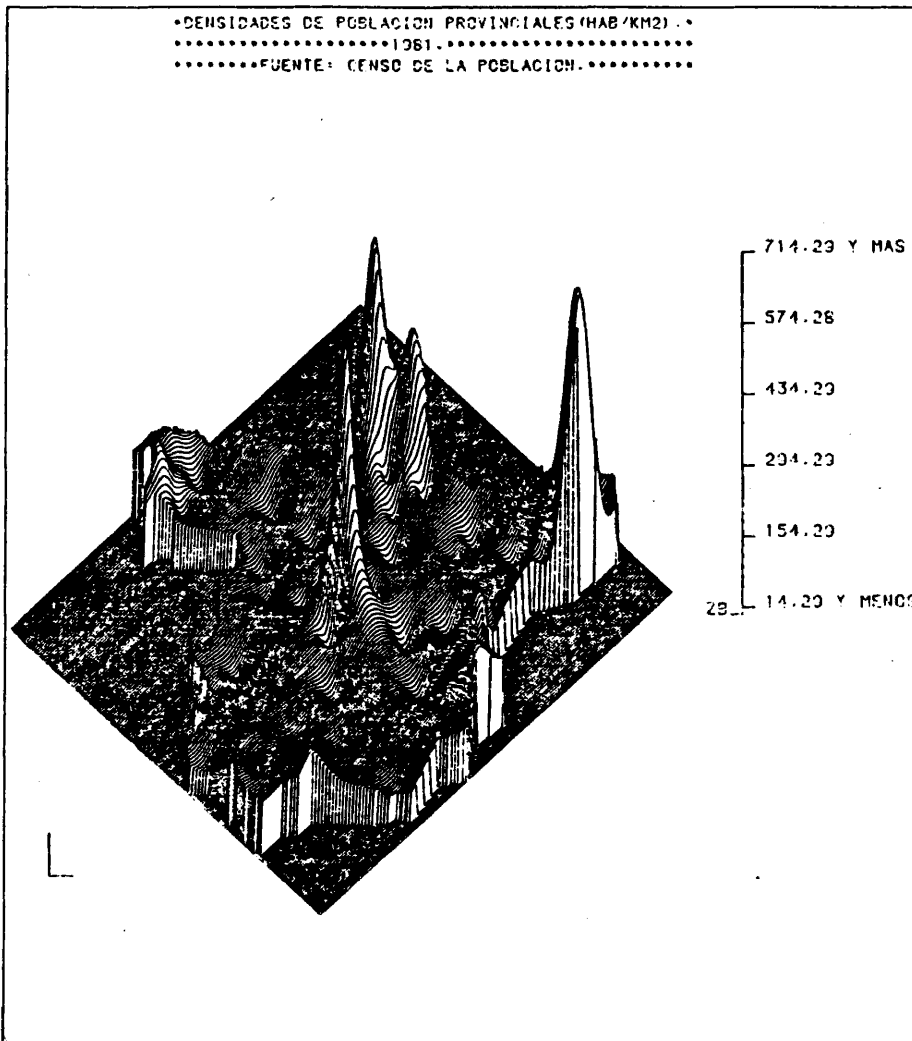


Fig.3.18.

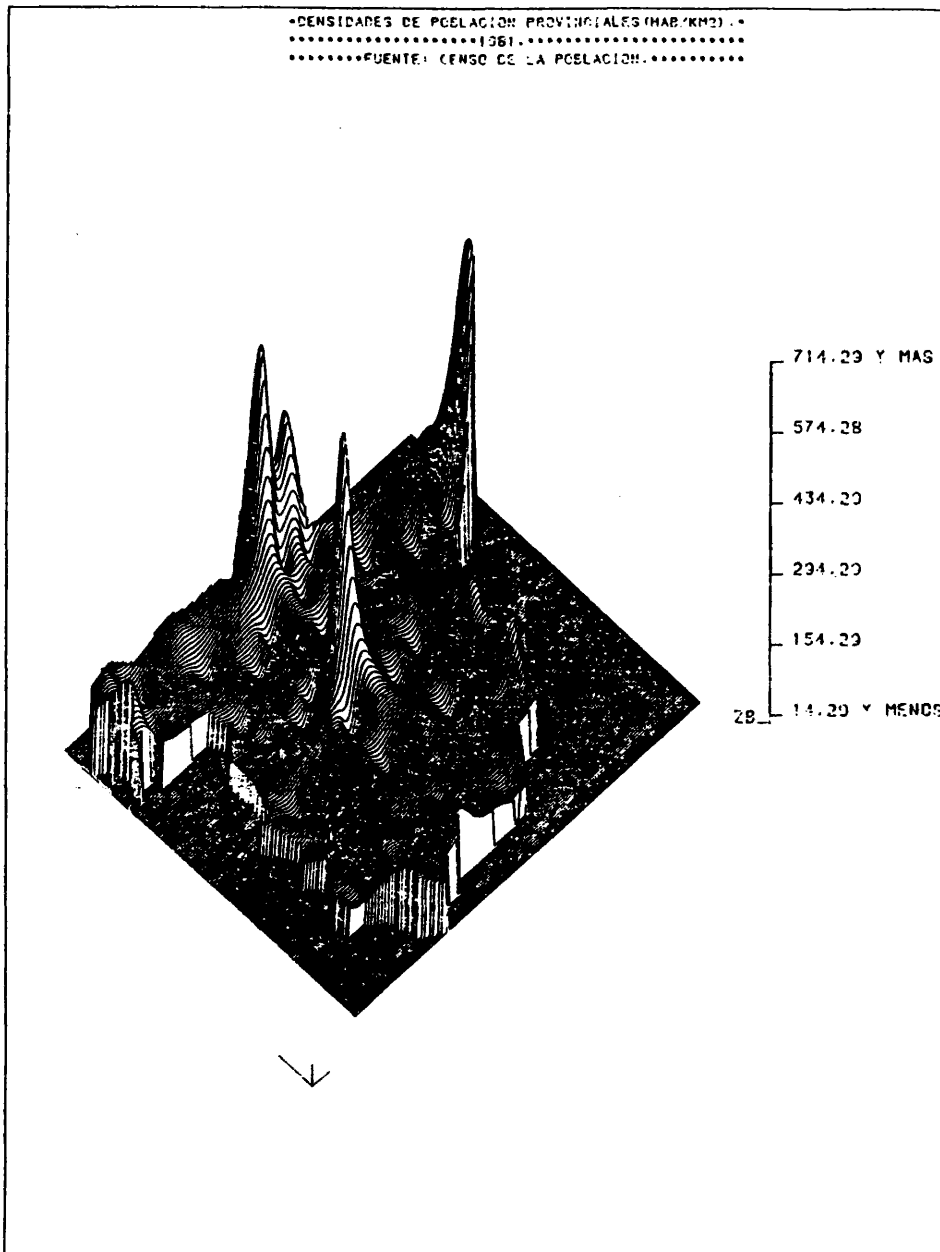


Fig.3.19.

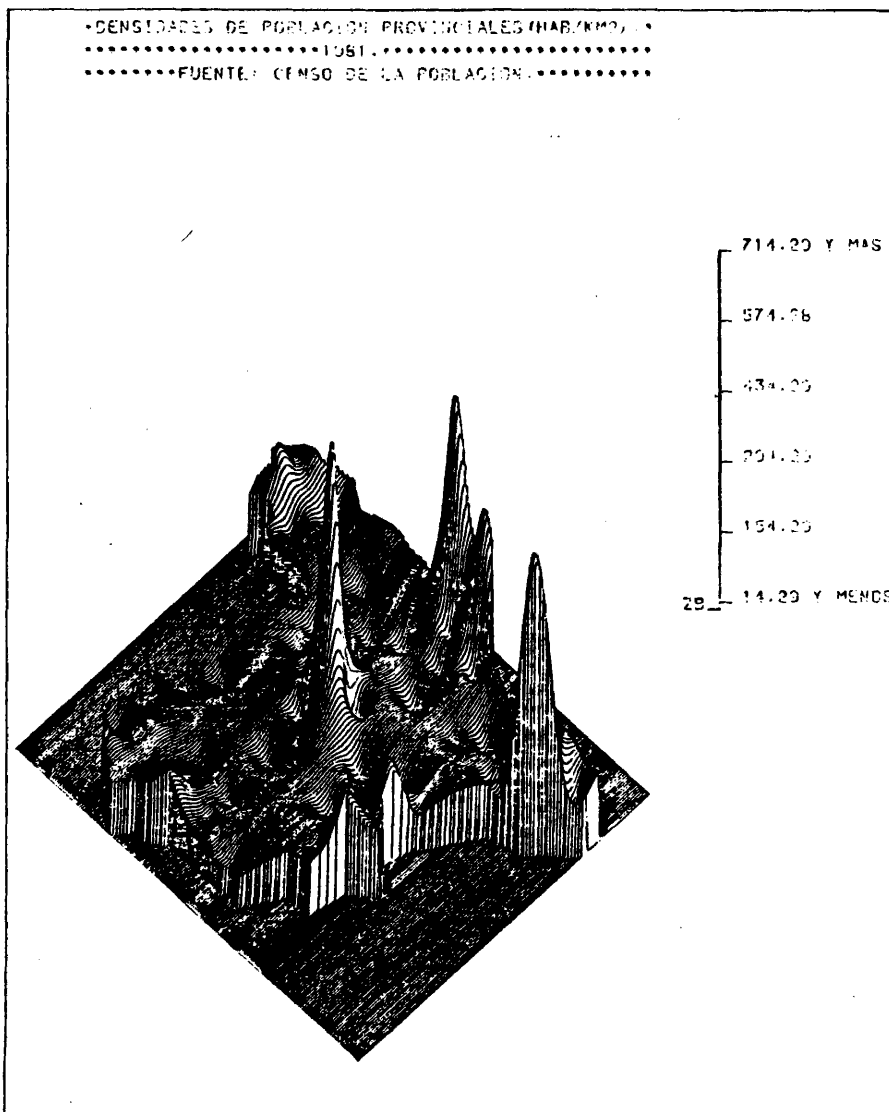


Fig.320.

```

*****
* PROGRAMA DE DIBUJO DE CONTORNOS EN PERSPECTIVA ISOMETRICA PARA *
* FACILITAR (MEDIANTE SUPERPONIBLES) LA INTERPRETACION DE UN BLOQUE *
* DIAGRAMA COHERENTE CON ESTOS. *
*****
NOMBRE DEL PROGRAMA: MOSAIC03.
LENGUAJE: FORTRAN77.
AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
      DEPARTAMENTO DE GEOGRAFIA HUMANA.
      UNIVERSIDAD COMPLUTENSE (MADRID).
      JULIO 1982.
ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
MEMORIA: 136 KBYTES.
SUBROUTINAS NECESARIAS: VISTA, PLOT3, TRN32, PLOTMM.
*****
*****
ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
LA INFORMACION SE ORGANIZA EN DOS FICHEROS.
FICHERO FT05F001.
*****
PRIMERA TARJETA.
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
//FT05F001 DD *
SEGUNDA TARJETA.
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
      AAA      BBB
A = NUMERO DE CONTORNOS A DIBUJAR.
B = FACTOR DE ESCALA DEL DIBUJO.
TERCERA TARJETA.
1      10      20      30      40      50      70      80
COL. |.....|.....|.....|.....|.....|.....|.....|
      CC,CC DD,DD
(C,D) = ORIGEN DE COORDENADAS DURANTE EL DIBUJO.
CUARTA TARJETA.

```

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      EE.EE FF.FF GG.GG
  
```

```

C E = COORDENADA X DEL PUNTO DE VISTA.
C F = COORDENADA Y DEL PUNTO DE VISTA.
C G = COORDENADA Z DEL PUNTO DE VISTA.
  
```

QUINTA TARJETA (COMO EJEMPLO).

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      (2F5.3)
  
```

FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LAS COORDENADAS DE LOS PUNTOS DE CADA CONTORNO.

ULTIMA TARJETA.

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      /*
  
```

FICHERO FT08F001.  
\*\*\*\*\*

PRIMERA TARJETA.

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      //FT08F001 DD *
  
```

POR CADA CONTORNO.

SEGUNDA TARJETA.

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      AAA
  
```

A = NUMERO DE PUNTOS DEL CONTORNO.

TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y DE CADA UNO DE LOS PUNTOS -POR ESTE ORDEN-, CON EL FORMATO DESCRITO.

ULTIMA TARJETA.

```

C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
      /*
  
```

```

C*****
C*****
C*****
  
```

```

DIMENSION XYZ(3,500),A(2)
CHARACTER FMT(80)
COMMON/XYMM/XM2,YM2,XM1,YM1,FCT
INTEGER OUT
DATA IN,OUT/5,6/
XM1 = 1.E74
YM1 = 1.E74
XM2 = -1.E74
  
```

```

      YM2 =-1.E74
      READ (IN,100) NCONT,FCT
      WRITE (OUT,100) NCONT,FCT
100  FORMAT (5X,13,5X,F3.0)
      READ (IN,201) XOR,YOR
      WRITE (OUT,201) XOR,YOR
201  FORMAT(5X,2F6.2)
      READ (IN,200) VX,VY,VZ
      WRITE(OUT,200) VX,VY,VZ
200  FORMAT(5X,3F6.2)
      CALL PLOTS (0,0,6)
C
C  CALCULO DE LA MATRIZ DE TRANSFORMACION DE COORDENADAS EN EL ESPACIO
C  DE TRES DIMENSIONES, EN COORDENADAS EN EL ESPACIO BIDIMENSIONAL DE
C  DIBUJO.
C
      CALL VISTA(VX,VY,VZ)
      CALL FACTOR (FCT)
      CALL PLOT(XOR,YOR,-3)
      READ (IN,101) (FMT(I),I=1,80)
      WRITE (OUT,102) (FMT(I),I=1,80)
101  FORMAT (80A1)
102  FORMAT ('OEL FORMATO DE LOS DATOS ES = 'w 80A1)
      IN=8
      DO 1 I=1,NCONT
C
C  SE LEE EL NUMERO DE PUNTOS DE CADA CONTORNO.
C
      READ (IN,100) LCONT
      WRITE (OUT,100) LCONT
C
C  SE LEEN LAS COORDENADAS DE LOS PUNTOS DE CADA CONTORNO.
C
      READ (IN,FMT) ((XYZ(I1,J1),I1=1,2),J1=1,LCONT)
      WRITE (OUT,103) ((XYZ(I1,J1),I1=1,2),J1=1,LCONT)
103  FORMAT (8X,'X',9X,'Y',/(5X,F6.2,4X,F6.2))
      DO 3 K=1,LCONT
3    XYZ(3,K)=0.
      IP=3
      N=LCONT+1
      XC=0.
      YC=0.
      DO 2 J=1,N
      X=XYZ(1,J)
      Y=XYZ(2,J)
      Z=XYZ(3,J)
      IF(J.E0,N) X=XYZ(1,1)
      IF(J.E0,N) Y=XYZ(2,1)
      IF(J.E0,N) Z=XYZ(3,1)
C
C  SE DIBUJA EL CONTORNO EN EL ESPACIO IMAGEN.
C
      CALL PLOT3(X,Y,Z,IP,0,0.)
      IP=2
      CALL TRN32(A,XYZ(1,J))
      XC=XC+A(1)
      YC=YC+A(2)
2    CONTINUE

```

```
XC=XC/N
YC=YC/N
R1=FLOAT(1)
C
C SE SITUA EL NUMERO DE ORDEN DEL CONTORNO EN EL INPUT SOBRE SU CEN-
C TROIDE EN EL ESPACIO IMAGEN.
C
CALL NUMBER(XC-0.1,YC-0.1,0.2,R1,0.,-1)
1 CONTINUE
CALL PLOT(-3.,-3.,9995)
C
C .....
C
STOP
END
```



### 3.5.2. El programa TREND.

Este programa, que constituye una versión alternativa del programa DIABLOCK, del que tan sólo se diferencia en el tipo de interpolación que realiza, construye automáticamente un bloque diagrama en perspectiva isométrica, que representa la variación espacial continua de una característica en el interior de un único contorno geográficamente significativo.

#### 3.5.2.1. Tipo de interpolación que contempla y datos que requiere, coherentemente, para su funcionamiento.

El programa TREND, en su estado actual, depende de los resultados de la utilización previa de algún programa de ajuste de superficies de tendencia a una muestra de datos espaciales, ya que parte de la lectura de los coeficientes de dicha superficie.

El ajuste de superficies de tendencia a una muestra de datos espaciales consiste en la definición de un polinomio,  $Z=f(X,Y)$ , que describa satisfactoriamente la relación que existe, en el conjunto de los puntos de la muestra, entre sus coordenadas en el plano horizontal  $(X,Y)$  y el valor de la característica  $(Z)$  en cada uno de ellos.

Los programas que realizan este tipo de tareas suelen efectuar ajustes de polinomios de complejidad creciente -es decir, de grado progresivamente más elevado- ofertando para cada caso un test de la bondad de ajuste, que normalmente es una función de la varianza.

De esta manera se dispone de un criterio para decidir qué tipo de superficie resulta más adecuada. Normalmente convendrá elegir la superficie que conlleve un incremento sustancial en la porción de varianza explicada respecto de la superficie de grado inmediatamente anterior, y que no sea manifiestamente desventajosa en este sentido respecto de las superficies de grado superior.

En todos los casos, los programas de ajuste de superficies de tendencia calculan e imprimen los valores de los coeficientes de cada superficie, ya que son éstos los parámetros que las definen, distinguiéndolas de todas las posibles de igual grado.

Supongamos que se ha ajustado una superficie de tendencia de tercer grado a los datos de una muestra espacial. Esta superficie tendrá la siguiente expresión polinómica:

$$/3.19/ Z = a+bX+cY+dX^2+eXY+fY^2+gX^3+hX^2Y+iXY^2+jY^3,$$

que es común a todos los polígonos  $Z = f(X,Y)$  de tercer grado. La superficie quedará definida, por tanto, por el valor concreto de los coeficientes:  $a, b, c, d, e, f, g, h, i, j$ , que afectan a cada uno de los términos del polinomio.

Pues bien, conocidos el grado y los coeficientes de la super

ficie de tendencia que describe adecuadamente la variación de una característica en el interior de un contorno geográficamente significativo como resultado de algún programa de ajuste, la subrutina de interpolación ZETA2, llamada desde TREND, averigua el valor de la característica en cada punto interior al contorno en cuestión de la retícula que se define para la construcción del volumen temático, calculando el valor del polinomio (Z) correspondiente a las coordenadas X e Y de dicho punto.

Los datos que requiere, por tanto, el programa TREND para su funcionamiento son los siguientes: grado y coeficiente de la superficie que se ha ajustado previamente, y valores de las coordenadas de los puntos que definen el contorno en consideración.

#### 3.5.2.2. Listado del programa y requisitos de uso.

Como puede apreciarse en el texto del programa que aparece a continuación, la estructura del mismo es muy semejante a la del programa DIABLOCK. Las únicas diferencias aparecen en la sección de lectura de datos de partida ya que en este caso, en vez de leer la localización de los centroides y el peso de la característica sobre ellos, se leen el grado y los coeficientes de la superficie ajustada, y en la llamada a la subrutina de interpolación -ZETA en el programa DIABLOCK, ZETA2 en TREND-.

Los requisitos de uso de TREND aparecen claramente expuestos en la documentación que precede al texto del programa propiamente dicho.

Los textos de las subrutinas, que a excepción de ZETA2 son las mismas que utiliza DIABLOCK, aparecen en el apéndice final, clasificados por orden alfabético.

```

C
C *****
C * PROGRAMA DE DIBUJO DE BLOQUES DIAGRAMA EN PERSPECTIVA ISOMETRI--
C *CA, QUE EXPRESAN LA VARIACION DE UNA CARACTERISTICA EN EL INTERIOR*
C *DE UN AREA, A PARTIR DE LOS COEFICIENTES DE UNA SUPERFICIE DE TEN-*
C *DENCIA PREVIAMENTE AJUSTADA A VALORES DE LA CARACTERISTICA EN UNA *
C *MUESTRA ESPACIAL INTERIOR A LA ZONA.
C * EL PROGRAMA NO REALIZA EL AJUSTE. PARTE DE LA LECTURA DEL GRADO*
C *DE LA SUPERFICIE Y DE LOS VALORES DE LOS COEFICIENTES DE ESTA. *
C * PARA UTILIZAR ESTE PROGRAMA, POR TANTO, ES NECESARIO REALIZAR *
C *PREVIAMENTE EL AJUSTE, RECURRIENDO A ALGUNO DE LOS PROCEDIMIENTOS *
C *INCLUIDOS EN LA MAYORIA DE LAS BIBLIOTECAS DE PROGRAMAS ESTADISTI-*
C *COS (BMDP, SPSS, ETC.).
C *
C * *****
C * EL PROGRAMA PUEDE CONSIDERAR HASTA UNA SUPERFICIE DE GRADO 6 Y *
C *UN CONTORNO DEFINIDO POR UN MAXIMO DE 100 PUNTOS. SI SE REBASAN *
C *ESTOS LIMITES ES NECESARIO REDIMENSIONAR LOS ARRAYS SUP,B,X E Y. *
C *****
C
C NOMBRE DEL PROGRAMA: TREND.
C
C LENGUAJE: FORTRAN77.
C
C AUTORES: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C MIGUEL GARCIA FERRANDEZ.
C DEPARTAMENTO DE ECUACIONES FUNCIONALES.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M.180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C
C MEMORIA: 192 KBYTES.
C
C SUBROUTINAS NECESARIAS:ZETA2, RAY, REDRG, SETG2, G1, G2, VISTA, PLOT3
C TRN32, CACHE, INTER, MERGE, INTP, SCVERT, TITULO, MARCO, PLOTMM.
C
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN TRES FICHEROS.
C
C FICHERO FT05F001 (REUNE TODOS LOS PARAMETROS DE CONTROL).
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
CBL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.

```

C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 AAAA BBBB CCCC.CCCC

- A = COORDENADA X DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO QUE DIAGRAMA.
- B = COORDENADA Y DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO QUE DIAGRAMA.
- C = FACTOR DE ESCALA DEL BIBUJO.

TERCERA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 DDDD EEEE FFFF GGGGG HHH.HHH III.III

- D = COORDENADA X DEL PUNTO DE VISTA ELEGIDO.
- E = COORDENADA Y DEL PUNTO DE VISTA ELEGIDO.
- F = COORDENADA Z DEL PUNTO DE VISTA ELEGIDO.
- G = ANGULO (EN EL SENTIDO DE LAS AGUJAS DEL RELOJ) QUE SE DESEA QUE FORME EL RAYADO CON LA VISUAL DIRIGIDA DESDE EL PUNTO DE VISTA AL ORIGEN DE COORDENADAS.
- H = VALOR Y DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGONO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.
- I = VALOR X DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGONO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.

CUARTA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 JJJJ KKKK.KKKK LLLL.LLLL MMMM.MMMM NNNN.NNNN

- J = VALOR DE LA FUNCION FUERA DEL CONTORNO.
- K = VALOR MINIMO DE LA FUNCION DENTRO DEL CONTORNO.
- L = VALOR MAXIMO DE LA FUNCION DENTRO DEL CONTORNO.
- M = FACTOR DE CORRECCION DE LA FUNCION (\*).
- N = FACTOR DE CORRECCION DE LA FUNCION (+).

QUINTA TARJETA (COMO EJEMPLO).  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 (8F10.0)

DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS VALORES DE LOS COEFICIENTES DE LA SUPERFICIE DE TENDENCIA.

SEXTA TARJETA (COMO EJEMPLO).  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 (8F10.0)

DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 9 LOS VALORES DE LAS COORDENADAS DE LOS PUNTOS QUE DEFINEN EL CONTORNO EXTERIOR DEL AREA.

SEPTIMA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|  
 0





```

202 FORMAT(5X,3F5.0,F6.0,2F8.3)
   READ(5,203) ZB,Z0,Z1,ZF,ZINC
   WRITE(6,203) ZB,Z0,Z1,ZF,ZINC
203 FORMAT(5X,F5.0,3F10.4,T40,F10.4)
C
C   LECTURA DEL GRADO Y DE LOS VALORES DE LOS COEFICIENTES DE LA SUPER-
C   FICIE AJUSTADA.
C
   READ(5,107)(FMT1(I),I=1,80)
   READ(8,888) NSUP
888 FORMAT(I3)
   NS1=NSUP+1
   READ(8,FMT1)((SUP(I,J),J=1,I),I=1,NS1)
   DO 500 I=1,NS1
500 WRITE(6,501)(SUP(I,J),J=1,I)
501 FORMAT(//('X',10F10.5))
C
C   LECTURA DE LAS COORDENADAS DEL CONTORNO EXTERIOR.
C
   READ(9,100) N
   WRITE(6,100) N
100 FORMAT(2X,I3)
   READ(5,107) FMT1
107 FORMAT(80A1)
   WRITE(6,104) FMT1
104 FORMAT('OFORMATO DEL CONTORNO = ',80A1/)
   READ(9,FMT1) (B(1,I),B(2,I),I=1,N)
   WRITE(6,105) N,(B(1,I),B(2,I),I=1,N)
105 FORMAT(I3/(2F10.3))
C
C*****
C*****
C*****
C*****
C
C   SECCION DE DIBUJO.
C   *****
C
C   SE ABRE EL DIBUJO.
C
   CALL PLOTS(0,0,6)
C
C*****
C
C   SE CONSIDERA EL FACTOR DE ESCALA.
C
   CALL FACTOR(FCT)
C
C   SE DESPLAZA EL ORIGEN DE COORDENADAS.
C
   CALL PLOT(XOR,YOR,-3)
C
C   SE CALCULA LA MATRIZ DE TRANSFORMACION DE COORDENADAS EN EL ESPACIO
C   DE TRES DIMENSIONES, EN COORDENADAS EN EL ESPACIO BIDIMENSIONAL DE
C   DIBUJO.
C
   CALL VISTA(VX,VY,VZ)
C

```

```

C SE PINTA UN TRIEDRO EN EL ORIGEN DE COORDENADAS.
C
CALL PLOT3(1.,0.,0.,3)
CALL PLOT3(0.,0.,0.,2)
CALL PLOT3(0.,2.,0.,2)
CALL PLOT3(0.,0.,3.,3)
CALL PLOT3(0.,0.,0.,2)
C
C SE DEFINE EL ANGULO DE RAYADO COHERENTE CON EL PUNTO DE VISTA ELE-
C GIDO Y CON LA PETICION MANIFESTADA POR EL USUARIO.
C
CALL SETG2(VX,VY,WRA)
WRA=-WRA
WR=WR-90.
WR=WR+WRA
C
C SE GIRA EL POLIGONO, SE ENMARCA Y SE CALCULA EL NUMERO DE RAYAS NE-
C CESARIAS PARA REPRODUCIR EL BLOQUE DIAGRAMA, EN FUNCION DE LA RETI-
C CULA ORTOGONAL DEFINIDA POR EL USUARIO AL DAR VALORES A DR Y A DS.
C
XMIN=1.E74
XMAX=-1.E74
YMAX=-1.E74
YMIN=1.E74
DO 1 I=1,N
CALL G1(X(I),Y(I),B(1,I),B(2,I),WR)
IF(YMAX.LT.Y(I)) YMAX=Y(I)
IF(YMIN.GT.Y(I)) YMIN=Y(I)
IF(XMAX.LT.X(I)) XMAX=X(I)
IF(XMIN.GT.X(I)) XMIN=X(I)
1 CONTINUE
WRITE(6,106) YMIN,YMAX,XMIN,XMAX
WRITE(6,106) (X(I),Y(I),I=1,N)
106 FORMAT(/(10X,2F10.3))
M=(YMAX-YMIN)/DR+1.5
YR=YMIN-(DR+0.06789)
C
C*****
C M3M=0
C MAX=600
C*****
C EN ESTA ITERACION SE DIBUJA EL BLOQUE DIAGRAMA.
C
C*****
C DO 20 J=1,M
C*****
C 20.1. SE RAYA EL POLIGONO EN TRES DIMENSIONES.
C
YR=YR+DR
CALL RAY(X,Y,N,YR,A,K)
IF(K.LE.0) GO TO 20
I=0
DX=DS
XR=XMIN-DX
A(K+1)=999999.
ICONT=0

```



```

2 CONTINUE
XR=XR+DX
IF(XR.GT.XMAX+DX) GO TO 5
IF(XR.LE.A(I+1)) GO TO 3
I=I+1
3 CONTINUE
CALL G2(XN,YN,XR,YR,WR)
Z=ZB
IF(MOD(I,2).EQ.0) GO TO 4
C
C 20.1.1. LLAMADA A LA SUBROUTINA DE INTERPOLACION.
C
CALL ZETA2(XN,YN,P,NS1,SUP,7)
IF(IDBS.GE.NDIS)
*WRITE(6,999) XR,YR,XN,YN,P
999 FORMAT(60X,5F10.3)
Z=ZF*(P+ZINC)
Z=AMAX1(Z0,Z)
Z=AMIN1(Z1,Z)
4 CONTINUE
C
C 20.2. SE CALCULAN LAS COORDENADAS DE LA RAYA EN EL ESPACIO IMAGEN.
C
ICONT=ICONT+1
LINEAA(1,ICONT)=XN
LINEAA(2,ICONT)=YN
LINEAA(3,ICONT)=Z
CALL TRN32(LINEA(1,ICONT),LINEAA(1,ICONT))
GO TO 2
5 CONTINUE
C
C 20.3. SE CALCULAN LAS SECCIONES NO OCULTADAS POR SECCIONES ANTERIO-
C RES, Y SE DIBUJAN.
C
IER=0
CALL CACHE(V,M3M,M33,MAX,LINEA,ICONT,W,LW,0,IER,0)
M3M=M33
C
IF(IER.GE.6) STOP
C
C*****
20 CONTINUE
C*****
C
C*****
C
SE DEVUELVE EL ORIGEN DE COORDENADAS AL PUNTO EN QUE SE ENCONTRABA
ANTES DE REALIZAR EL DIBUJO DEL BLOQUE DIAGRAMA.
CALL PLOT(-XOR,-YOR,-3)
C
C*****
C
EL DIBUJO DEL DIAGRAMA TRIDIMENSIONAL ESTA COMPRENDIDO ENTRE LOS
VERTICES (XM1,YM1) Y (XM2,YM2).
XM1=XM1+XOR*FCT
YM1=YM1+YOR*FCT

```

```

XM2=XM2+XDR*FCT
YM2=YM2+YDR*FCT
C
C*****
C
C LLAMADA A LA SUBROUTINA QUE SITUA LA ESCALA VERTICAL DE LECTURA DEL
C DIAGRAMA TRIDIMENSIONAL A LA DERECHA DEL DIBUJO.
C
  XSC=(XM2/FCT)+3.
  YSC=(YM2/FCT)/2.
  CALL SCVERT(XSC,YSC,ZB,ZO,ZI,ZF,ZINC,XTOP,YTOP)
  XM2=AMAX1(XM2,(XTOP*FCT))
  YM2=AMAX1(YM2,(YTOP*FCT))
C
C*****
C
C LLAMADA A LA SUBROUTINA QUE REPRODUCE EL TITULO DEL DIAGRAMA.
C
  CALL FACTOR(1.)
  CALL TITULO(XM1,YM2,(XM2-XM1),4.)
  YM2=YM2+4.
C
C*****
C
C SE ENMARCA EL DIBUJO.
C
  CALL PLOT(0.,0.,3)
  XMARC=XM1-1.
  YMARC=YM1-1.
  CALL MARCD(XMARC,YMARC,(XM2+1.-XM1),(YM2+1.-YM1))
C
C*****
C
C SE CIERRA EL DIBUJO.
C
  CALL PLOT(9.,-8.,999)
C
C*****
C
  STOP
  END

```

\*\*\*PREFERENCIAS ESPACIALES EN LA ESPAÑA PENINSULAR (SUPERFICIE DE TENDENCIA\*\*\*  
\*\*\*\*\*DE GRADO 3). FUENTE: ENCUESTA REALIZADA POR D. JOSE ESTEBANEZ.\*\*\*\*\*

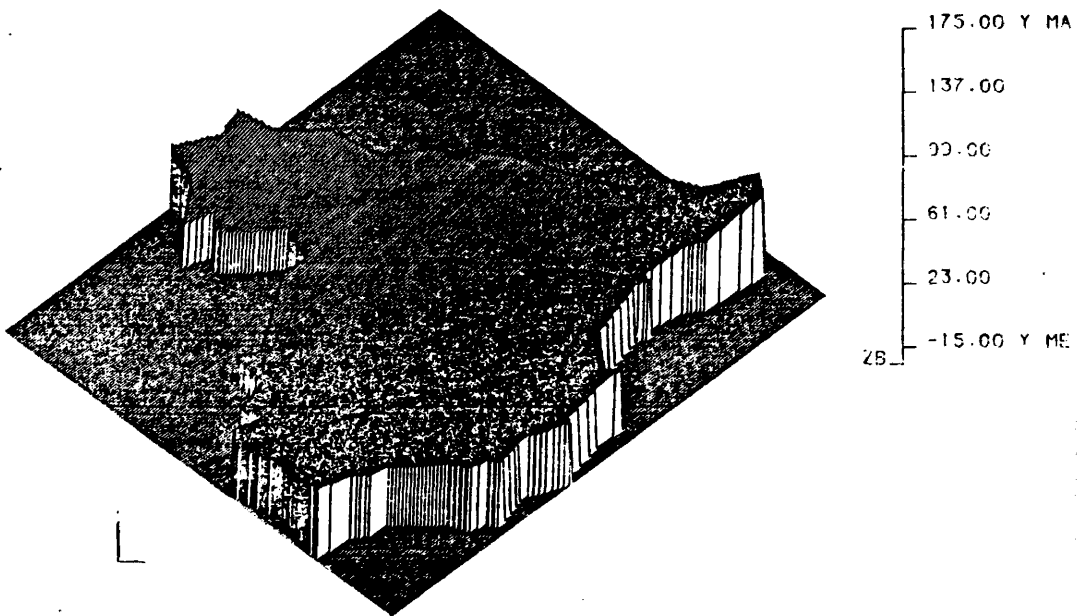


Fig. 3.21.

•PREFERENCIAS ESPACIALES EN LA ESPAÑA PENINSULAR (SUPERFICIE DE TENDENCIA•••••  
•••••DE GRADO 4). FUENTE: ENCUESTA REALIZADA POR D. JOSE ESTEBANEZ.\*\*\*\*\*

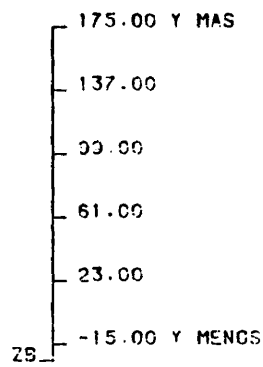


Fig.3.22.

\*\*\*PREFERENCIAS ESPACIALES EN LA ESPAÑA PENINSULAR (SUPERFICIE DE TENDENCIA\*\*\*  
\*\*\*\*\*DE GRADO 5): FUENTE: ENCUESTA REALIZADA POR D. JOSE ESTEBANEZ.\*\*\*\*\*

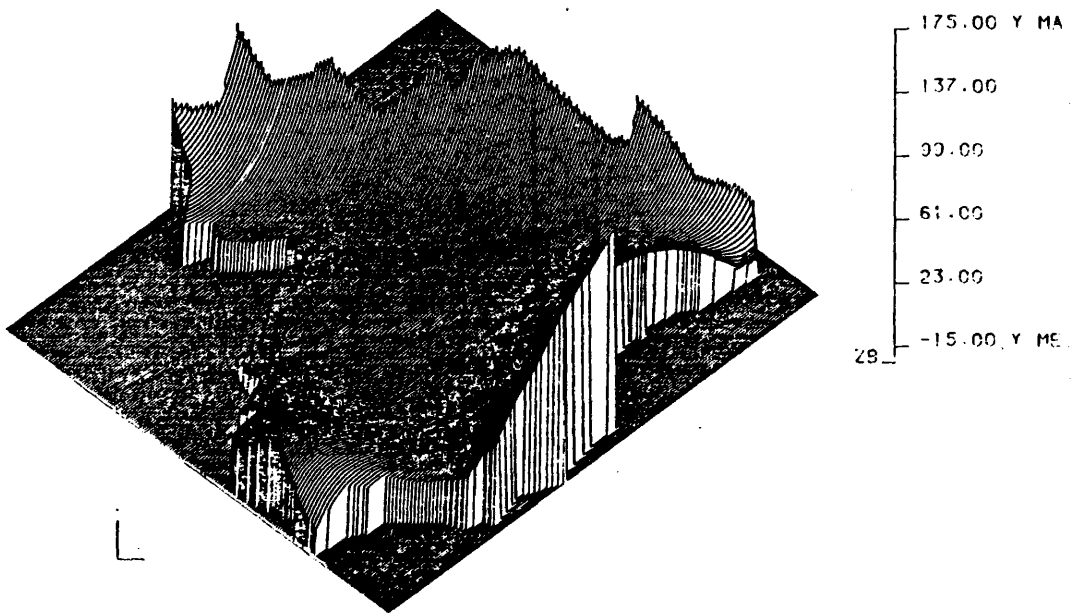


Fig. 3.23.

•PREFERENCIAS ESPACIALES EN LA ESPAÑA PENINSULAR (SUPERFICIE DE TENDENCIA•••••  
•••••DE GRADO 6). FUENTE: ENCUESTA REALIZADA POR D. JOSE ESTEBANEZ. •••••

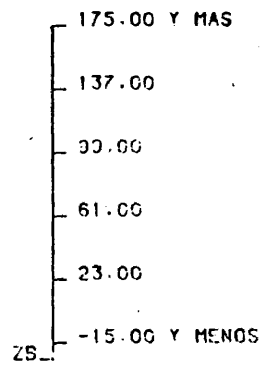


Fig 3.24.

### 3.5.2.3. Algunos ejemplos.

Los gráficos de las figuras 3.21., 3.22., 3.23., y 3.24., son el resultado de la utilización de TREND en la obtención de bloques diagrama que representan preferencias espaciales de residencia en la España peninsular (4).

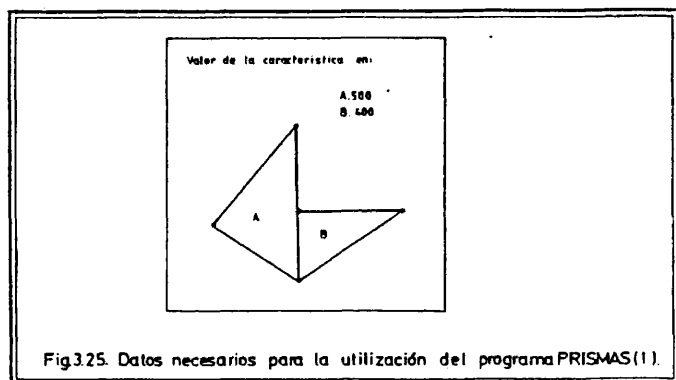
### 3.5.3. El programa PRISMAS.

Este programa proporciona una vista en perspectiva isométrica de un volumen temático en el que se consideran diversos contornos geográficamente significativos, asumiendo que el peso de la característica es invariable en el interior de cada uno de ellos.

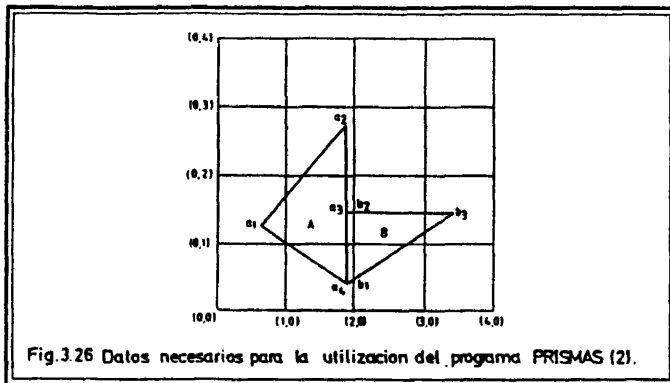
Los procedimientos de dibujo son básicamente los mismos que se utilizan en los programas DIABLOCK y TREND. No ocurre lo mismo, en cambio, con la naturaleza de los datos que requiere el programa, ni con el procedimiento de interpolación a que se invoca.

#### 3.5.3.1. Tipo de interpolación que contempla y datos que requiere, coherentemente, para su funcionamiento.

Comencemos por considerar la información necesaria para el funcionamiento del programa, suponiendo que se trata de obtener un bloque diagrama de la distribución de una característica sobre dos espacios, definidos por sus contornos (Fig. 3.25.).



Superponemos un retículo ortogonal y definimos un origen de coordenadas (Fig. 3.26.).



Los dos contornos quedan descritos de la siguiente manera:

Contorno A	X	Y
$a_1$	0.6	1.3
$a_2$	1.8	2.9
$a_3$	1.8	1.5
$a_4$	1.8	0.6

Contorno B	X	Y
$b_1$	1.8	0.6
$b_2$	1.8	1.5
$b_3$	3.5	1.5

Por otra parte, construimos una lista en la que aparecen los valores de la característica en el interior de cada contorno, en el mismo orden en que han sido definidos éstos.

	P
Contorno A (1°)	500
Contorno B (2°)	400



Una vez definida esta referencia espacial -coordenadas de los puntos que aproximan cada contorno-, para la construcción del relieve temático se asume que el peso de la variable es constante en el interior de cada contorno. Coherentemente con ello, el algoritmo de interpolación investiga a qué polígono es interior cada uno de los puntos significativos para la representación del volumen, suponiendo sobre cada uno de ellos un peso equivalente al que afecta a los polígonos a los que son interiores:

$$/3.20/ \quad z_i = z_k \Leftrightarrow P_i \in A_k$$

Donde:

$z_i$  = valor interpolado de la característica en el punto  $i$ .

$z_k$  = valor de la característica en el polígono  $k$ .

$P_i$  = punto  $i$ .

$A_k$  = polígono (área)  $k$ .

El resultado gráfico es un conjunto de prismas levantados sobre los contornos, de altura proporcional al peso de la variable en éstos.

Vemos, por tanto, que la información necesaria para el funcionamiento del programa está constituida por dos conjuntos de datos. En primer lugar, es necesario conocer numéricamente -por la lista de las coordenadas de los puntos que los aproximan- cada uno de los contornos que se van a considerar. A continuación, y en el mismo orden en que han sido definidos espacialmente, se introducen los valores de la característica en cuestión sobre cada uno de ellos.

### 3.5.3.2. Listado del programa y requisitos de uso.

En el texto que aparece a continuación destaca en primer lugar la alusión a un conjunto de subrutinas específicas de este programa, que no son compartidas por los programas DIABLOCK o TREND: ORDEN, CHECK2, CONCE2, AFIN2 y AFINE. La razón de ello es que este programa, a diferencia de los dos anteriores, establece un rayado sobre más de un polígono, siendo imprescindible por ello disponer de tratamientos complementarios, que no son necesarios cuando sólo se considera un contorno.

Por otra parte, en un examen más detenido del texto, se puede comprobar que la definición de puntos significativos para la representación del relieve que realiza este programa, difiere de la que efectúan los otros dos procedimientos de dibujo de bloques diagrama.

En los programas DIABLOCK y TREND, los puntos significativos para la descripción del relieve se encuentran uniformemente

distribuidos, a lo largo de un retículo ortogonal, ajustado a las dimensiones del contorno de la base. Los puntos de corte de las líneas horizontales del retículo con el contorno se utilizan sólo para reconocer qué puntos significativos son exteriores a éste y, por ello, en qué puntos se anula la característica.

En el programa PRISMAS, en cambio, los puntos significativos para el dibujo del diagrama son, precisamente, los puntos de corte de las líneas horizontales del retículo -que ya no es tal, sino un haz de paralelas- con todos los contornos que se consideran.

Sólo se reproduce ahora el texto del programa principal. Los textos de todas las subrutinas que éste utiliza se encuentran en el apéndice final.

```

C
C *****
C * PROGRAMA DE DIBUJO DE PRISMAS EN PERSPECTIVA ISOMETRICA, LEVAN-
C * TADOS SOBRE CONTORNOS GEOGRAFICAMENTE SIGNIFICATIVOS, DE ALTURA
C * PROPORCIONAL AL PESO DE UNA CARACTERISTICA EN ELLOS.
C * *****
C * EL PROGRAMA PUEDE CONSIDERAR HASTA 50 AREAS DISTINTAS. SI SE RE-
C * BASA ESTE LIMITE ES NECESARIO REDIMENSIONAR LOS ARRAYS AREAS, LONG*
C * , PES, A, PDLINT Y LONGA.
C * *****
C
C NOMBRE DEL PROGRAMA: PRISMAS.
C
C LENGUAJE: FORTRAN77.
C
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C
C MEMORIA: 240 KBYTES.
C
C SUBRUTINAS NECESARIAS: RAY, REORG, SETG2, G1, G2, VISTA, PLOT3,
C TRN32, ORDEN, CHECK2, CONCE2, AFIN2, AFINE, CACHE, INTER, MERGE,
C INTP, SCVERT, TITULO, MARCO, PLOTMM.
C
C *****
C *****
C *****
C
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C
C LA INFORMACION SE ORGANIZA EN TRES FICHEROS.
C
C FICHERO FT05F001 (REUNE TODOS LOS PARAMETROS DE CONTROL).
C *****
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C AAAA BBBB CCCC.CCCC
C
C A = COORDENADA X DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO
C QUE DIAGRAMA.
C B = COORDENADA Y DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLO
C QUE DIAGRAMA.
C C = FACTOR DE ESCALA DEL BIBUJO.

```

C  
C TERCERA TARJETA.  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C DDDD EEEE FFFF GGGGG HHH,HHH  
C  
C D = COORDENADA X DEL PUNTO DE VISTA ELEGIDO.  
C E = COORDENADA Y DEL PUNTO DE VISTA ELEGIDO.  
C F = COORDENADA Z DEL PUNTO DE VISTA ELEGIDO.  
C G = ANGULO (EN EL SENTIDO DE LAS AGUJAS DEL RELOJ) QUE SE DESEA  
C QUE FORME EL RAYADO CON LA VISUAL DIRIGIDA DESDE EL PUNTO DE  
C VISTA AL ORIGEN DE COORDENADAS.  
C H = SEPARACION DE LAS RAYAS QUE COMPONEN EL DIBUJO.  
C  
C CUARTA TARJETA.  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C JJJJ KKKK,KKKK LLLL.LLLL MMMM,MMMM NNNN.NNNN  
C  
C J = VALOR DE LA FUNCION FUERA DEL CONTORNO.  
C K = VALOR MINIMO DE LA FUNCION DENTRO DEL CONTORNO.  
C L = VALOR MAXIMO DE LA FUNCION DENTRO DEL CONTORNO.  
C M = FACTOR DE CORRECCION DE LA FUNCION (\*).  
C N = FACTOR DE CORRECCION DE LA FUNCION (+).  
C  
C QUINTA TARJETA (COMO EJEMPLD).  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C (2F6,3)  
C  
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 8 LOS  
C VALORES DE LAS COORDENADAS DE LOS PUNTOS QUE DEFINEN LOS CONTORNOS  
C QUE SE VAN A CONSIDERAR.  
C  
C SEXTA TARJETA (COMO EJEMPLD).  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C (F10,0)  
C  
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 9 LOS  
C PESOS DE LA CARACTERISTICA SOBRE LOS CONTORNOS.  
C  
C SEPTIMA TARJETA.  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C 0  
C  
C 0 = NUMERO DE TARJETAS (MAXIMO 3) QUE FORMAN EL TITULO DEL DIBUJO.  
C  
C OCTAVA TARJETA (0 TARJETAS).  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C DISTRIBUCION ESPACIAL DE.....  
C  
C ULTIMA TARJETA.  
C 1 10 20 30 40 50 70 80  
C COL. |.....|.....|.....|.....|.....|.....|.....|  
C /\*  
C

```

C
C FICHERO FT08F001 (NUMERO DE CONTORNOS ,NUMERO DE PUNTOS QUE DESCRIBEN
C ***** BEN CADA CONTORNO Y VALORES DE LAS COORDENADAS DE
C LOS PUNTOS).
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C AAA
C
C A = NUMERO DE AREAS ( O CONTORNOS).
C
C POR CADA AREA.
C
C TERCERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C BBB
C
C B = NUMERO DE PUNTOS DEL CONTORNO.
C
C CUARTA TARJETA Y SIGUIENTES (TANTAS COMO PUNTOS).
C
C COORDENADA X, COORDENADA Y DE CADA PUNTO, CON EL FORMATO DESCRITO
C PREVIAMENTE.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT09F001 (PESOS DE LA CARACTERISTICA SOBRE LAS AREAS EN CON-
C ***** SIDERACION).
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C //FT09F001 DD *
C
C TANTAS TARJETAS COMO AREAS.
C
C EN CADA UNA DE ELLAS, EL PESO DE LA CARACTERISTICA EN EL AREA
C CORRESPONDIENTE, CON EL FORMATO DESCRITO PREVIAMENTE.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
C COL. ....|.....|.....|.....|.....|.....|.....|
C /*
C
C *****
C *****
C *****
C

```

```

COMMON/XYMM/XM2,YM2,XM1,YM1,FCT
DIMENSION AREAS(100,15),LONG(50),PES(50),AREAA(2)
DIMENSION A(8,50),SINTEX(300,3)
DIMENSION W(3,800),V(2,800)
DIMENSION AREASA(15,2)
CHARACTER FMT(80),FMT1(80)
REAL LINEA(2,400),LINEAA(3,400),MEZCLA(300,3)
LOGICAL LW(800)
INTEGER POLINT(50),OUT,LONGA(50)

C
C AREAS = MATRIZ DONDE SE ALMACENAN LAS COORDENADAS DE LOS PUNTOS
C           QUE DESCRIBEN LOS CONTORNOS.
C LONG = VECTOR EN EL QUE SE ALMACENAN LOS DIFERENTES NUMEROS DE
C           PUNTOS QUE DESCRIBEN LOS CONTORNOS.
C PES = VECTOR DE PESOS DE LA VARIABLE SOBRE LOS CONTORNOS.
C FMT,FMT1 = VECTORES DE CARACTERES PARA LA DESCRIPCION DE LOS FORMA-
C           TOS DE LOS DATOS QUE APORTA EL USUARIO.
C AREAA,A,SINTEX,W,V,AREASA,LINEA,LINEAA,MEZCLA,LW,POLINT,LONGA = PO-
C           SICIONES DE MEMORIA QUE SE UTILIZAN PARA EL CALCULO DE TODOS
C           LOS VALORES NUMERICOS NECESARIOS PARA LA CONSTRUCCION DEL
C           DIBUJO.
C (XM2,YM2) = COORDENADAS DEL VERTICE SUPERIOR DERECHO DEL MINIMO REC-
C           TANGULO QUE ENMARCA EL DIBUJO.
C (XM1,YM1) = COORDENADAS DEL VERTICE INFERIOR IZQUIERDO DEL MINIMO
C           RECTANGULO QUE ENMARCA EL DIBUJO.
C
C *****
C
C INICIALIZACION DE VARIABLES DE CONTROL DE LECTURA Y ESCRITURA.
C
C   DATA INDIC/0/
C   DATA IN,OUT/5,6/
C
C INICIALIZACION DE VALORES DE LA ZONA COMUN XYMM PARA AVERIGUAR EL
C RECUADRO.
C
C   XM1 = 1.E74
C   XM2 = -1.E74
C   YM1 = 1.E74
C   YM2 = -1.E74
C
C *****
C
C LECTURA DE DATOS.
C *****
C
C LECTURA DE PARAMETROS.
C
C   READ(IN,201)XOR,YOR,FCT
C   WRITE(OUT,201)XOR,YOR,FCT
C 201 FORMAT(5X,2F5.0,F10.4)
C   READ(IN,202)VX,VY,VZ,WR,DR,DS
C   WRITE(OUT,202)VX,VY,VZ,WR,DR,DS
C -202 FORMAT(5X,3F5.0,F6.0,2F8.3)
C   READ(IN,203)ZB,Z0,Z1,ZF,ZINC
C   WRITE(OUT,203)ZB,Z0,Z1,ZF,ZINC
C 203 FORMAT(5X,F5.0,4F10.4)
C

```

```

C SE DEFINÉ EL ANGULO DE RAYADO COHERENTE CON EL PUNTO DE VISTA ELE-
C GIDO Y CON LA PETICION MANIFESTADA POR EL USUARIO.
C
  CALL SETG2(VX,VY,WRA)
  WRA=-WRA
  WR=WR-90.
  WR=WR+WRA
C
C LECTURA DEL NUMERO DE CONTORNOS QUE SE CONSIDERAN.
C
  IN=8
  READ(IN,204)NCONT
  WRITE(OUT,204)NCONT
204 FORMAT(5X,I3)
C
C LECTURA DE CONTORNOS Y SUS LONGITUDES. GIRO DE TODOS LOS PUNTOS,
C SEGUN EL ANGULO DE RAYADO, Y BUSQUEDA DE LOS MAXIMOS Y MINIMOS,
C PARA ENCUADRAR EL DIBUJO. LA Y MAXIMA Y LA Y MINIMA DE CADA
C CONTORNO SE ALMACENAN PARA DETERMINAR FACILMENTE QUE CONTORNOS
C SON CORTADOS POR CADA LINEA DE DIBUJO.
C
  XMIN=1.E74
  XMAX=-1.E74
  YMIN=1.E74
  YMAX=-1.E74
  IN=5
  READ(IN,208)FMT
208 FORMAT(80A1)
  WRITE(OUT,209)FMT
209 FORMAT(/'OFORMATO DE LOS CONTORNOS = '280A1)
  IN=8
  NCONT2=NCONT*2
  I1=0
  DO 1 I=1,NCONT2,2
  I1=I1+1
  READ(IN,205)LONG(I1)
205 FORMAT(5X,I3)
  WRITE(OUT,206)I1
206 FORMAT(/'0CONTORNO ',I3/'0COORDENADAS'/9X,'X',6X,'Y')
  LON=LONG(I1)
  DO 2 J=1,LON
  READ(IN,FMT)AREAA(1),AREAA(2)
  WRITE(OUT,207)AREAA(1),AREAA(2)
207 FORMAT(5X,2F7.2)
  CALL G1(AREAS(I,J),AREAS(I+1,J),AREAA(1),AREAA(2),WR)
  IF(XMAX.LT.AREAS(I,J)) XMAX=AREAS(I,J)
  IF(XMIN.GT.AREAS(I,J)) XMIN=AREAS(I,J)
  IF(YMAX.LT.AREAS(I+1,J)) YMAX=AREAS(I+1,J)
  IF(YMIN.GT.AREAS(I+1,J)) YMIN=AREAS(I+1,J)
  2 CONTINUE
  1 CONTINUE
  IF(INDIC.NE.0)
  *WRITE(OUT,210)YMIN,YMAX,XMIN,XMAX
210 FORMAT(/'OYMIN=',F10.3,5X,'YMAX=',F10.3,5X,'XMIN=',F10.3,5X,'XMAX=
  *',F10.3)
  I1=0
  DO 3 I=1,NCONT2,2
  I1=I1+1

```

```

LON=LONG(I1)
IF(INDIC.NE,0)
*WRITE(OUT,211) (AREAS(I,J),AREAS(I+1,J),J=1,LON)
3 CONTINUE
211 FORMAT(6(6X,'X',9X,'Y',3X)/(12F10.3))
NI=0
DO 77 I=2,NCONT,2
NI=NI+1
YMINI=1.E74
YMAXI=-1.E74
LON=LONG(NI)
DO 78 J=1,LON
IF(YMAXI.LT,AREAS(I,J)) YMAXI=AREAS(I,J)
78 IF(YMINI.GT,AREAS(I,J)) YMINI=AREAS(I,J)
AREAS(I,LON+1)=YMINI
AREAS(I,LON+2)=YMAXI
77 CONTINUE
C
C LECTURA DE LOS PESOS DE LA VARIABLE EN LOS DIFERENTES CONTORNOS.
C
IN=5
READ(IN,208) FMT1
WRITE (OUT,212) FMT1
212 FORMAT(/'FORMATO DE LOS PESOS DE LA VARIABLE = ',80A1)
IN=9
DO 4 I=1,NCONT
4 READ(IN,FMT1) PES(I)
WRITE(OUT,213) (I,PES(I),I=1,NCONT)
213 FORMAT(5X,I3,F10.3)
C
C *****
C *****
C *****
C *****
C
C SECCION DE DIBUJO.
C *****
C
C SE ABRE EL DIBUJO.
C
CALL PLOTS(0,0,6)
C
C *****
C
C SE CONSIDERA EL FACTOR DE ESCALA.
C
CALL FACTOR(FCT)
C
C SE DESPLAZA EL ORIGEN DE COORDENADAS.
C
CALL PLOT(XOR,YOR,-3)
C
C SE CALCULA LA MATRIZ DE TRANSFORMACION DE COORDENADAS EN EL ESPACIO
C DE TRES DIMENSIONES, EN COORDENADAS EN EL ESPACIO BIDIMENSIONAL DE
C DIBUJO.
C
CALL VISTA(VX,VY,VZ)
C

```



```

C SE PINTA UN TRIEDRO EN EL ORIGEN DE COORDENADAS.
C
CALL PLOT3(1.,0.,0.,3)
CALL PLOT3(0.,0.,0.,2)
CALL PLOT3(0.,2.,0.,2)
CALL PLOT3(0.,0.,3.,3)
CALL PLOT3(0.,0.,0.,2)
C
C SE CALCULA LA COORDENADA Y DE LA PRIMERA RAYA Y EL NUMERO DE RAYAS
C NECESARIAS PARA REPRODUCIR EL DIBUJO.
C
M=(YMAX-YM|N)/DR+1.5
YR=YM|N-DR-0.067891
C
C*****
C M3M=0
C MAX=800
C*****
C
C EN ESTA ITERACION SE DIBUJAN LOS PRISMAS.
C
C*****
C DO 20 J=1,M
C*****
C YR=YR+DR
C
C 20.1. SE LOCALIZAN LOS CONTORNOS QUE INTERSECA CADA RAYA.
C
K1=0
I1=0
DO 21 J1=2,NCONT2,2
I1=I1+1
LON=LONG(I1)
IF((YR.LT.AREAS(J1,LON+1)).OR.(YR.GE.AREAS(J1,LON+2)))GOTO21
K1=K1+1
POLINT(K1)=I1
21 CONTINUE
IF(K1.LE.0) GOTO 20
C
C 20.2. SE CALCULAN AHORA LOS PUNTOS DE CORTE DE CADA CONTORNO CON LA
C RAYA DEL DIBUJO, Y SE ALMACENAN POR COLUMNAS EN LA MATRIZ A.
C
DO 22 JK=1,K1
I0=POLINT(JK)
LON=LONG(I0)
I033=2*I0-1
DO 422 KLM=1,LON
AREASA(KLM,1)=AREAS(I033,KLM)
422 AREASA(KLM,2)=AREAS(I033+1,KLM)
CALL RAY(AREASA(1,1),AREASA(1,2),LON,YR,A(1,JK),K1)
IF(INDIC.NE.0)
*WRITE(OUT,214) I0,(A(L1,JK),L1=1,K1)
22 LONGA(JK)=K1
214 FORMAT(I3/(2F10.3))
C
C 20.3. SE CONSTRUYE AHORA UNA LISTA EN LA QUE CADA FILA ESTA FORMADA
C POR TRES VALORES: COMIENZO DEL SEGMENTO, FINAL DEL SEGMENTO,

```

```

C VALOR DE LA VARIABLE EN ESE SEGMENTO.
C
  I42=0
  DO 23 JL=1,K1
  LONA=LONGA(JL)
  I3=POLINT(JL)
  DO 24 JM=1,LONA,2
  I42=I42+1
  MEZCLA(I42,1)=A(JM,JL)
  MEZCLA(I42,2)=A(JM+1,JL)
24 MEZCLA(I42,3)=PES(I3)
23 CONTINUE
  IF(INDIC.NE.0)
  *WRITE(OUT,815)((MEZCLA(J99,J98),J98=1,3),J99=1,I42)
815 FORMAT(5X,3F10.3)
  CALL ORDEN(MEZCLA,300,I42,3,2)
  IF(INDIC.NE.0)
  *WRITE(OUT,815)((MEZCLA(J99,J98),J98=1,3),J99=1,I42)
C
C 20.3.1. TRATAMIENTO DE LOS SEGMENTOS PERTENECIENTES A POLIGONOS
C CONCENTRICOS.
C
  CALL CHECK2(MEZCLA,300,I42,3)
  IF(INDIC.NE.0)
  *WRITE(OUT,815)((MEZCLA(J99,J98),J98=1,3),J99=1,I42)
C
  CALL AFIN2(MEZCLA,300,I42)
  IF(INDIC.NE.0)
  *WRITE(OUT,815)((MEZCLA(J99,J98),J98=1,3),J99=1,I42)
  DO 25 K20=1,I42
  DO 25 K21=1,3
25 SINTEX(K20,K21)=MEZCLA(K20,K21)
  LSINT=I42
  IF(INDIC.NE.0)
  *WRITE(OUT,792)((SINTEX(L42,L43),L43=1,3),L42=1,LSINT)
792 FORMAT(5X,3F10.3)
C
C 20.4. SE CONSTRUYE LA RAYA TRIDIMENSIONAL.
C
  LINEAA(1,1)=XMIN
  LINEAA(2,1)=YR
  LINEAA(3,1)=ZB
  I32=1
  DO 55 I33=1,LSINT
  LINEAA(1,I32+1)=SINTEX(I33,1)
  LINEAA(2,I32+1)=YR
  LINEAA(3,I32+1)=ZB
  LINEAA(1,I32+2)=SINTEX(I33,1)
  LINEAA(2,I32+2)=YR
  LINEAA(3,I32+2)=(SINTEX(I33,3)+2*INC)*ZF
  LINEAA(3,I32+2)=AMAX1(Z0,LINEAA(3,I32+2))
  LINEAA(3,I32+2)=AMIN1(Z1,LINEAA(3,I32+2))
  LINEAA(1,I32+3)=SINTEX(I33,2)
  LINEAA(2,I32+3)=YR
  LINEAA(3,I32+3)=(SINTEX(I33,3)+2*INC)*ZF
  LINEAA(3,I32+3)=AMAX1(Z0,LINEAA(3,I32+3))
  LINEAA(3,I32+3)=AMIN1(Z1,LINEAA(3,I32+3))
  LINEAA(1,I32+4)=SINTEX(I33,2)

```

```

LINEAA(2, I32+4)=YR
LINEAA(3, I32+4)=ZB
I32=I32+4
55 CONTINUE
LINEAA(1, I32+1)=XMAX
LINEAA(2, I32+1)=YR
LINEAA(3, I32+1)=ZB
I321=I32+1
IF(INDIC.NE.0)
*WRITE(OUT,793) ((LINEAA(KLM1,KLM2),KLM1=1,3),KLM2=1,I321)
CALL AFINE(LINEAA,I32+1,I49)
IF(INDIC.NE.0)
*WRITE(OUT,793) ((LINEAA(KLM1,KLM2),KLM1=1,3),KLM2=1,I49)
793 FORMAT(4(6X,'X',9X,'Y',9X,'Z',3X)/(12F10.3))
DO 79 I79=1,I49
CALL G2(XN,YN,LINEAA(1,I79),LINEAA(2,I79),WR)
LINEAA(1,I79)=XN
LINEAA(2,I79)=YN
79 CONTINUE
C
C 20.5. SE CONSTRUYE LA RAYA BIDIMENSIONAL, PREPARADA PARA EL DIBUJO.
C
C DO 56 I35=1,I49
56 CALL TRN32(LINEA(1,I35),LINEAA(1,I35))
C
C 20.6. SE CALCULAN LAS SECCIONES NO OCULTADAS POR SECCIONES ANTERIO-
C RES, Y SE DIBUJAN.
C
IER=0
CALL CACHE(V,M3M,M33,MAX,LINEA,I49,W,LW,0,IER,0)
IF (INDIC.NE.0)
*WRITE(OUT,795)
795 FORMAT('OV')
IF (INDIC.NE.0)
*WRITE(OUT,794) ((V(N50,N51),N50=1,2),N51=1,M33)
794 FORMAT('1'/5(6X,2F10.3))
M3M=M33
C
IF(IER.GE.6)STOP
C
C*****
20 CONTINUE
C*****
C
C.....
C
C SE DEVUELVE EL ORIGEN DE COORDENADAS AL PUNTO EN QUE SE ENCONTRABA
C ANTES DE REALIZAR EL DIBUJO DEL BLOQUE DIAGRAMA.
C
CALL PLOT(-XOR,-YOR,-3)
C
C.....
C
C EL DIBUJO DEL DIAGRAMA TRIDIMENSIONAL ESTÁ COMPRENDIDO ENTRE LOS
C VERTICES (XM1,YM1) Y (XM2,YM2).
C
XM1=XM1+XOR*FCT
YM1=YM1+YOR*FCT

```

```

      XM2=XM2+XOR*FCT
      YM2=YM2+YOR*FCT
C
C*****
C
C LLAMADA A LA SUBROUTINA QUE SITUA LA ESCALA VERTICAL DE LECTURA DEL
C DIAGRAMA TRIDIMENSIONAL A LA DERECHA DEL DIBUJO.
C
      XSC=(XM2/FCT)+3.
      YSC=(YM2/FCT)/2.
      CALL SCVERT(XSC,YSC,ZB,ZO,Z1,ZF,ZINC,XTOP,YTOP)
      XM2=AMAX1(XM2,(XTOP*FCT))
      YM2=AMAX1(YM2,(YTOP*FCT))
C
C*****
C
C LLAMADA A LA SUBROUTINA QUE REPRODUCE EL TITULO DEL DIAGRAMA.
C
      CALL FACTOR(1.)
      CALL TITULO(XM1,YM2,(XM2-XM1),4.)
      YM2=YM2+4.
C
C*****
C
C SE ENMARCA EL DIBUJO.
C
      CALL PLOT(0.,0.,3)
      XMARC=XM1-1.
      YMARC=YM1-1.
      CALL MARCO(XMARC,YMARC,(XM2+1.-XM1),(YM2+1.-YM1))
C
C*****
C
C SE CIERRA EL DIBUJO.
C
      CALL PLOT(9.,-8.,999)
C
C*****
C
      STOP
      END

```

#### 3.5.3.3. Algunos ejemplos.

Los dibujos que se presentan, constituyen vistas complementarias de un volumen que describe el valor de la densidad de población de las provincias de la España peninsular en 1981 (5). Como puede apreciarse, se trata del mismo fenómeno que se ha representado páginas arriba, utilizando el programa DIABLOCK. Por la comparación de ambos tipos de gráficos se pueden obtener conclusiones acerca de su mayor o menor adecuación a la representación de dichos fenómenos (ver Cebrián de Miguel y García Ferrández /35/).

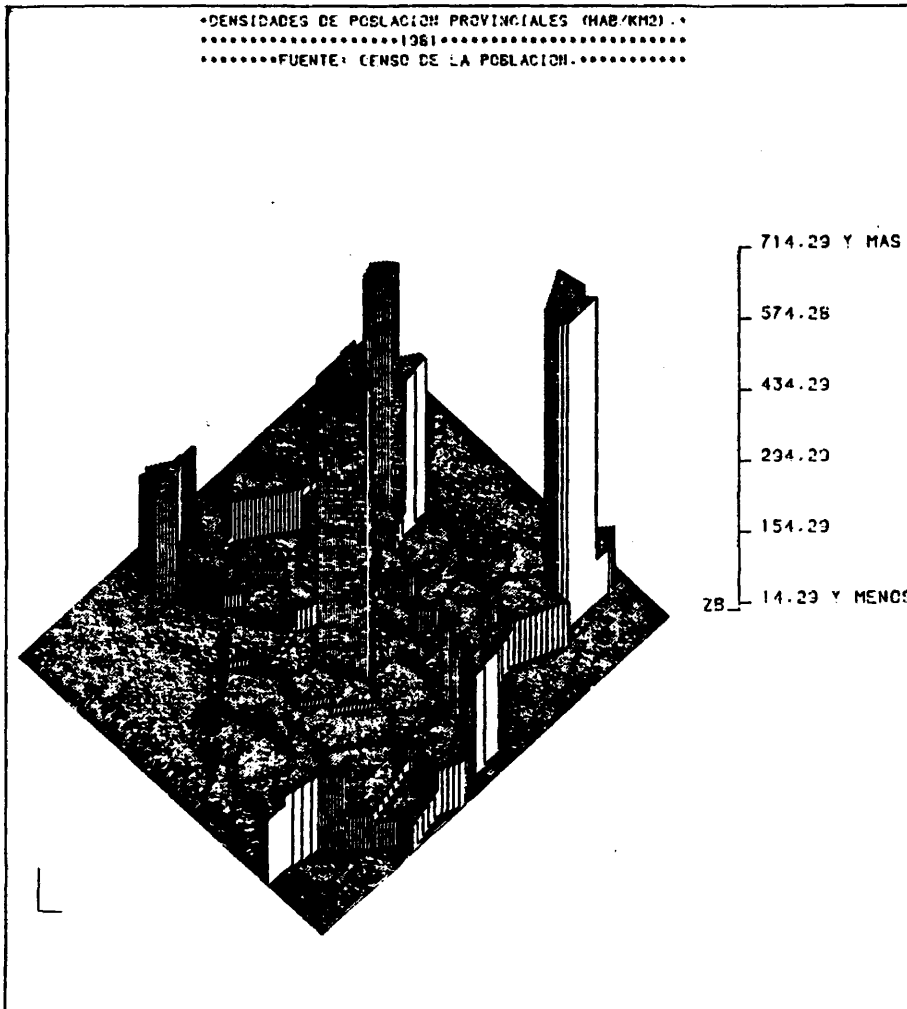


Fig.3.27.

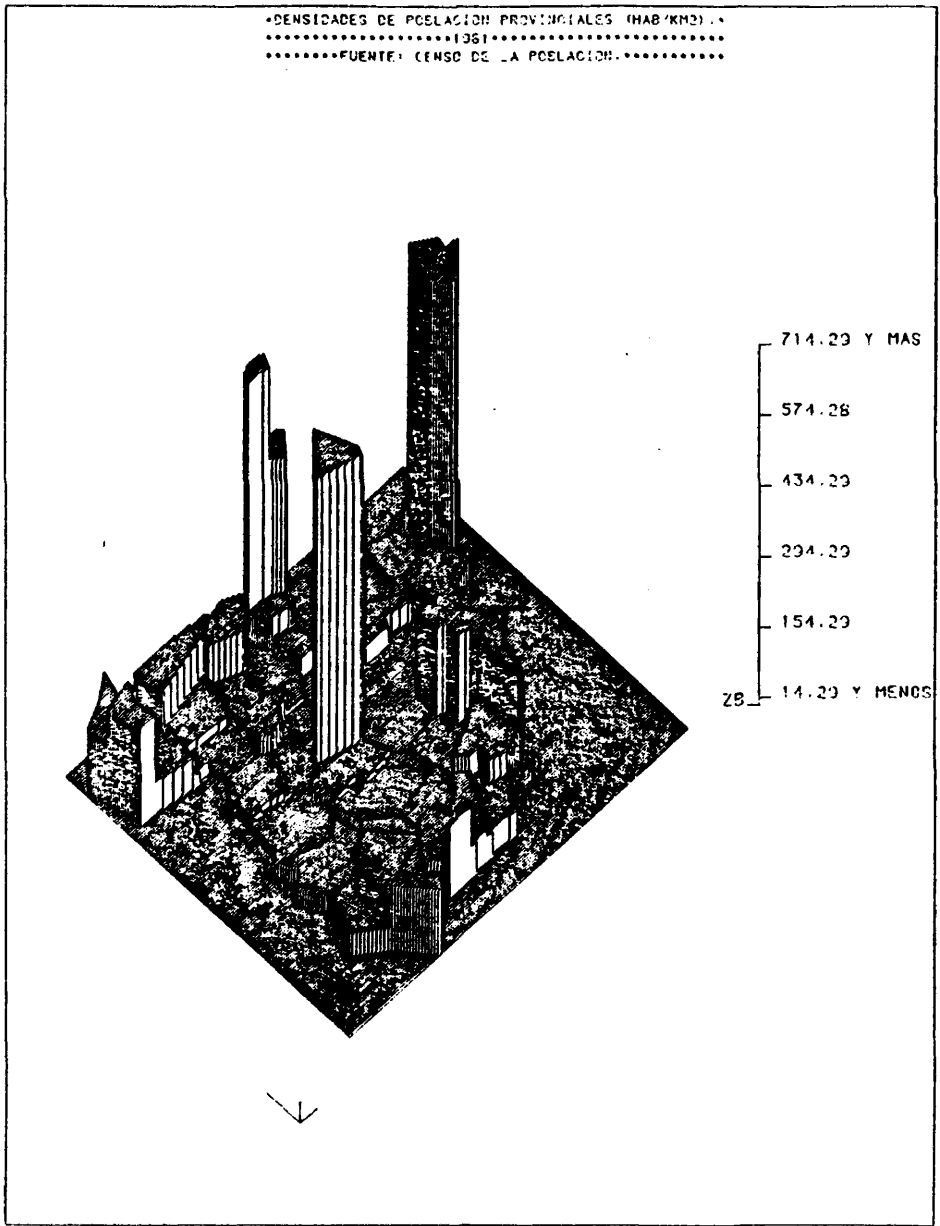


Fig.3.28.

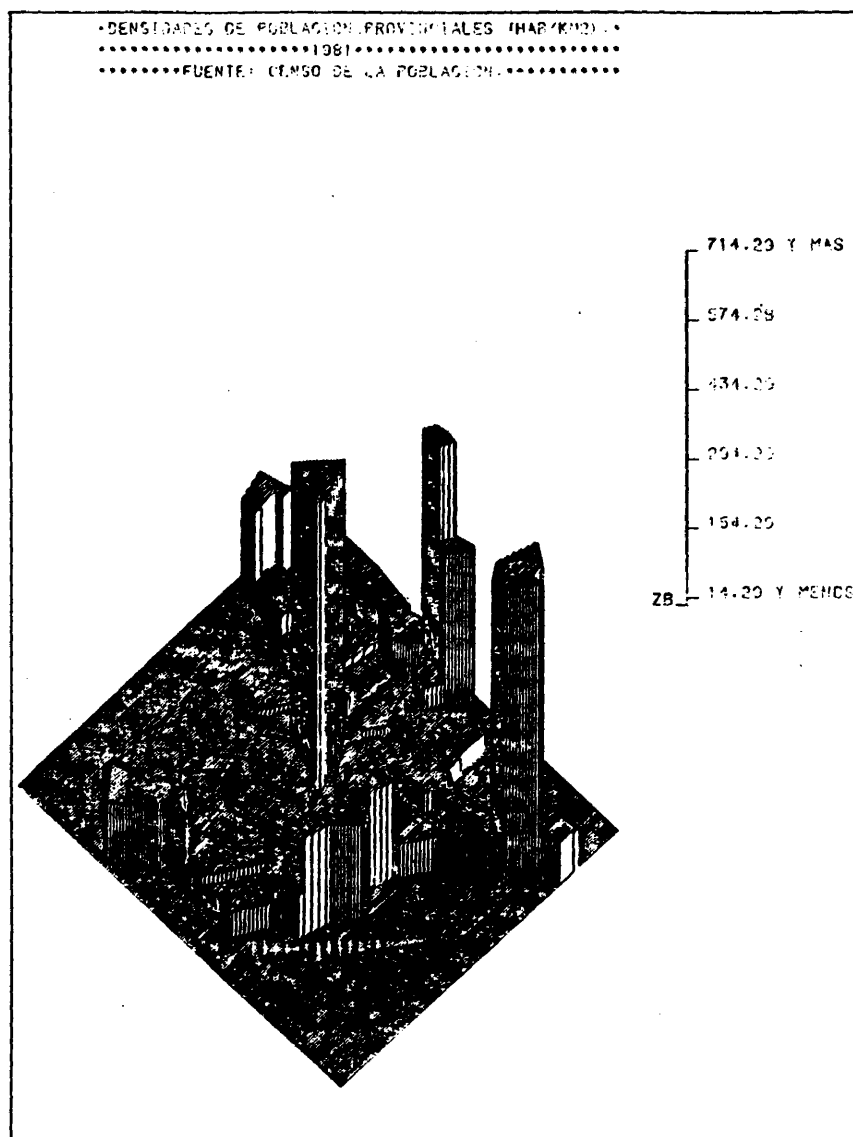


Fig.3.29.



NOTAS AL CAPITULO 3

- (1) Esta subrutina, definida como tal, se corresponde con una de las entradas de la subrutina PLOT3, diseñada por Miguel García Ferrández.
- (2) Esta subrutina, es, al igual que VISTA, otra de las entradas de la subrutina PLOT3, diseñada por Miguel García Ferrández.
- (3) Ver capítulo 2, apartado 2.3.3.
- (4) Los valores de los coeficientes de la superficie de tendencia han sido recogidos de los resultados del tratamiento, mediante programas desarrollados en la Universidad de Nottingham por Paul M. Mather, de datos procedentes de una encuesta realizada por el profesor Estébanez.
- (5) La descripción digital de los contornos de las provincias de la España peninsular me ha sido facilitada por Joaquín Bosque Sendra, que ya había utilizado estos datos en trabajos de cartografía automática mediante SYMAP.

CAPITULO 4. LA CUARTA DIMENSION: EL TIEMPO. ANIMACION GRAFICA  
DE BLOQUES DIAGRAMA.

---

	<u>Págs.</u>
4.1. INTRODUCCION .....	232
4.2. ANIMACION GRAFICA Y SERIES HISTORICAS .....	233
4.3. EL PROGRAMA FILM. ....	238
4.3.1. Descripción general de las tareas que realiza.	239
4.3.2. El problema de la interpolación temporal.....	245
4.3.3. Filmación de los resultados .....	249

#### 4.1. INTRODUCCION

Las primeras referencias acerca de la conveniencia de establecer procedimientos de cartografía dinámica se remontan a los primeros 60 (Thrower /163/ y /164/).

En ese momento inicial se proponen técnicas de elaboración manual de los gráficos que constituyen posteriormente los fotogramas de una secuencia animada.

Poco tiempo después, a mediados de la década de los 60, Corwell y Robinson /40/, plantean por primera vez las posibilidades de las técnicas de representación gráfica mediante ordenador en el establecimiento de secuencias animadas, ilustrativas de procesos de cambio espacio-temporal.

A partir de esa fecha y hasta mediados de los 70, se llevan a cabo múltiples realizaciones en este campo sobre temas muy diversos: procesos de erosión, cambios en la distribución de la población, crecimiento urbano (Tobler /168/), contaminación atmosférica, cambios climatológicos, procesos sísmicos, distribución de los accidentes de tráfico (Moellering /111/, /112/ y /113/), etc.

En la mayoría de los casos se sigue la estrategia de reproducir automáticamente cada fotograma, bien en una pantalla de rayos catódicos que es filmada bajo control, bien utilizando directamente un microfilm-plotter conectado a un ordenador.

Más recientemente, se ha comenzado a trabajar en el desarrollo de sistemas de cartografía dinámica en tiempo real (Moellering /115/) que permiten un control sobre la filmación de la escena mientras se está produciendo. Ello supone un avance importante, por cuanto permite obtener interactivamente la animación que resulta más expresiva, sin tener que decidir previamente todos los parámetros de control de la grabación, con los riesgos que ello lleva consigo.

El procedimiento que presentamos en este capítulo se sitúa en la línea de los desarrollados -en EEUU básicamente- en el período 1966-1976. La razón de ello es que para establecer un sistema de cartografía dinámica en tiempo real es necesario disponer de periféricos gráficos a los que no se ha podido acceder por el momento, y definir un conjunto de programas de tratamiento de la información gráfica sumamente complejos que, dada su inviabilidad actual, no nos ha parecido lógico redactar.

#### 4.2. ANIMACION GRAFICA Y SERIES HISTORICAS.

Para representar gráficamente la evolución temporal de un determinado fenómeno, se ha recurrido siempre a caracterizar las dos dimensiones ortogonales del plano de la siguiente manera: la componente horizontal recoge la escala de la evolución histórica y la componente vertical la de la magnitud del hecho en consideración.

De esta manera, la nube de puntos  $P = \{P_1, P_2, \dots, P_n\}$ , tal que  $P_i = (t_i, v_i)$ , siendo  $t_i =$  momento temporal y  $v_i =$  magnitud del fenómeno en el momento  $t_i$  (Fig. 4.1.), o la línea que une dichos puntos (Fig. 4.2.), representan adecuadamente, de forma discreta y continua, respectivamente, el proceso de cambio en cuestión.

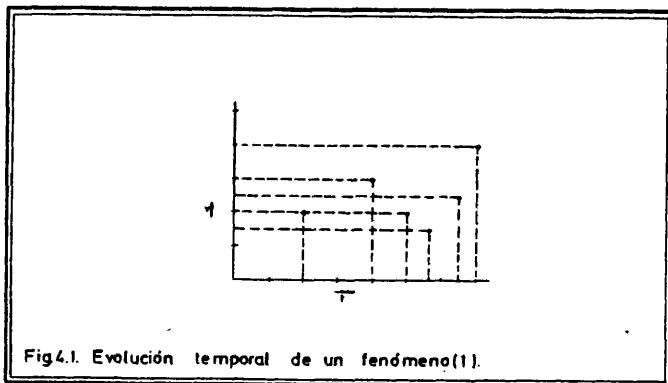
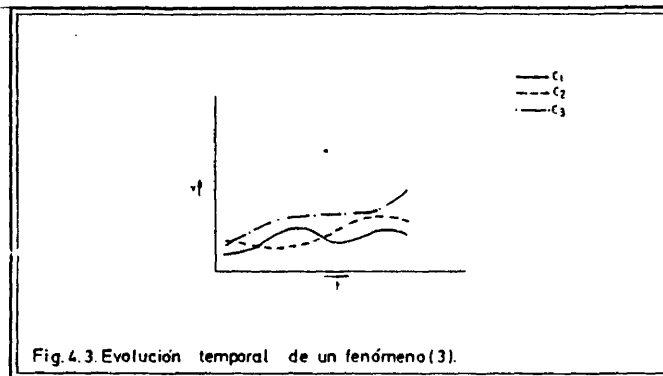
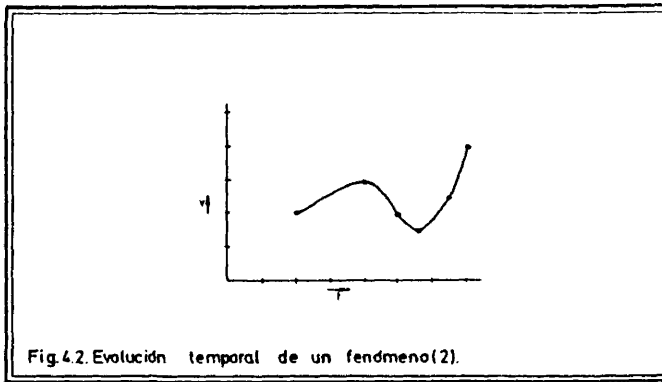


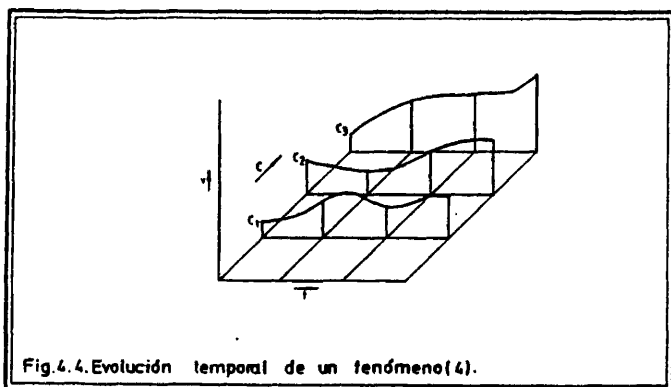
Fig.4.1. Evolución temporal de un fenómeno(1).

Este sistema de representación tiene limitaciones muy claras, ya que sólo puede ser utilizado, tal como ha sido descrito, en supuestos de evolución temporal de una determinada característica que afecta a un único individuo, o caso.

Si se pretende, en cambio, representar simultáneamente la evolución temporal de un determinado atributo que afecta a varios individuos, es necesario recurrir, bien a algún tipo de signo convencional -utilizando líneas de diferente trazado (Fig. 4.3.)-, bien a liberar una tercera dimensión en el plano de dibujo, a lo largo de la cual se alinean los diferentes casos (Fig. 4.4.).

La localización de los casos en las representaciones que hemos considerado es claramente espacial, ya que no ha tenido en cuenta su situación real en puntos significativos de la superficie terrestre. Si pretendemos ahora obtener una representación de la variación espacio-temporal de una determinada característica,





comprobaremos que no existe procedimiento satisfactorio de plasmación de este hecho en un único gráfico.

En efecto, si representamos el valor de la característica en puntos de una retícula espacial (Figs. 4.5. y 4.6.), se saturan inmediatamente las tres dimensiones lineales coherentes que podemos distinguir en el plano de dibujo.

Para obtener, por tanto, una representación gráfica de la variación espacio-temporal de una característica, es necesario recurrir a una sucesión de dibujos, cada uno de los cuales expresa la situación que se produce en un determinado momento temporal (Fig. 4.7.).

Este ha sido el modo normal, durante mucho tiempo, de presentación de series históricas de características espaciales, ya que la colección de mapas temáticos permite un estudio de la evolución temporal de la distribución espacial de un determinado hecho.

Los trabajos de cartografía dinámica, en cambio, suponen una aproximación más acertada a la plasmación de evoluciones temporales de variables espaciales.

El fundamento de este tipo de tratamientos es muy simple. La razón de su complejidad radica en dificultades puramente técnicas.

Si filmamos, manteniendo invariables los ejes de referencia, una colección de gráficos mínimamente desemejantes, representando cada uno de ellos un momento temporal de la distribución de una característica en una muestra espacial, y los presentamos a la vis

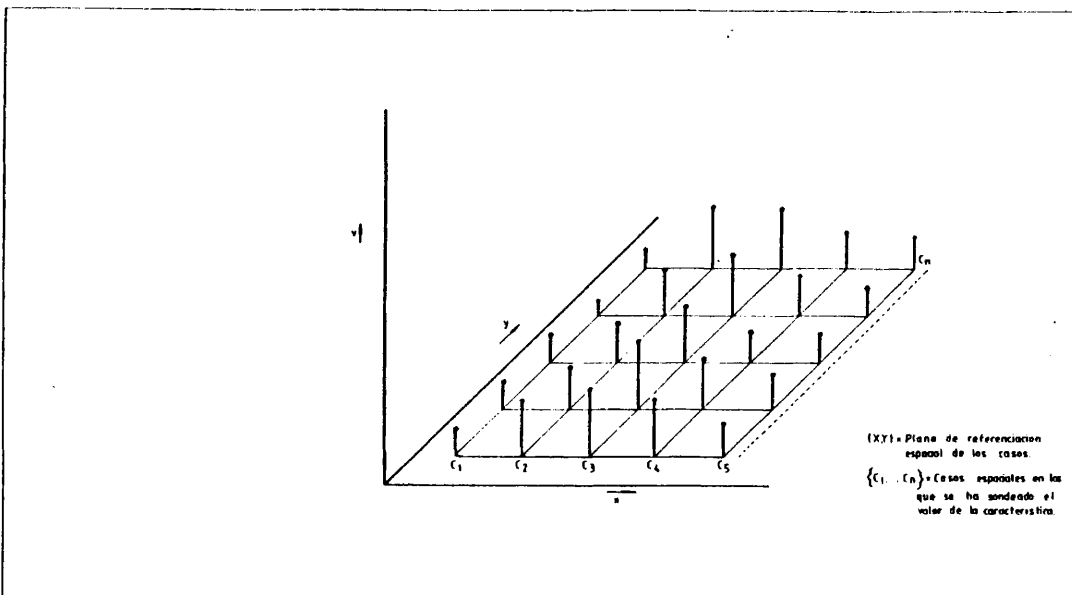


Fig.4.5. Representación espacial de una variable (I).

ta a la velocidad de 24 imágenes por segundo, el resultado será la animación continuada de un volumen -si se ha optado, como en el caso de la presente tesis, por un tipo de representación en bloques diagrama vistos en perspectiva isométrica- cuyas sucesivas configuraciones explicitan, con una fuerza expresiva cualitativamente distinta, las pautas fundamentales de un determinado proceso de evolución espacio-temporal.

La primera dificultad que es necesario superar para obtener una secuencia animada es el establecimiento de la colección, muy numerosa, de imágenes para la filmación. Si no se dispone de procedimientos automáticos de diseño, este escollo es prácticamente insalvable.

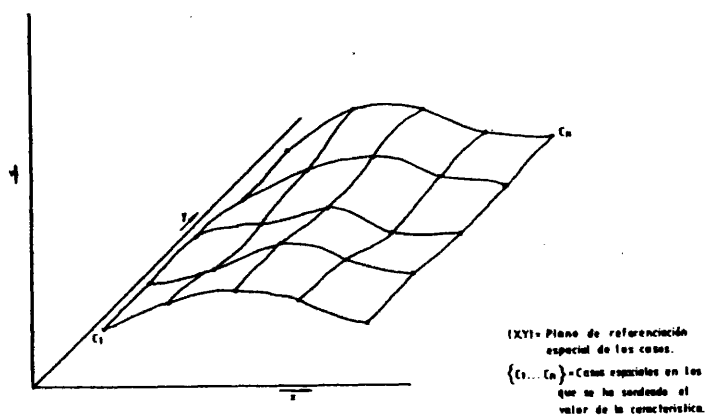


Fig.4.6. Representación espacial de una variable (2).

La solución del problema de la invariabilidad a lo largo de toda la filmación de los ejes básicos de referencia espacial, también resulta prácticamente insalvable si no se dispone de mecanismos automáticos de control.

En el trabajo que presentamos en este capítulo ha sido posible solucionar el primer problema, no así el segundo, por no haber tenido acceso a las máquinas adecuadas.



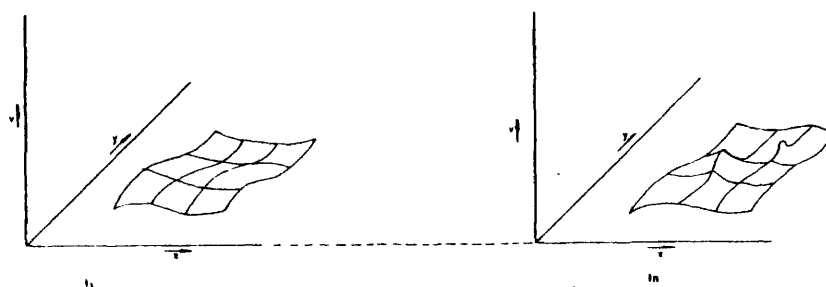


Fig4.7. Representación de la variación temporal de una característica espacial.

---

#### 4.3. EL PROGRAMA FILM.

Este programa ha sido diseñado para obtener automáticamente un gran número de imágenes, muy próximas entre sí, que representen, cada una de ellas, la distribución de una característica espacial en el interior de un contorno geográficamente significativo, en un momento temporal dado.

Posteriormente, los resultados gráficos del funcionamiento de este programa pueden ser filmados para obtener una secuencia animada de una dinámica espacio-temporal.

#### 4.3.1. Descripción general de las tareas que realiza.

El programa parte de la lectura de los siguientes datos, que debe introducir el usuario:

- . Coordenadas de los puntos que describen el contorno en cuyo interior se va a representar la distribución espacial de una característica.
- . Valores de las coordenadas de los centroides interiores al contorno, en los cuales se ha sondeado el valor de la característica a representar en dos momentos temporales.
- . Valores de la característica sobre los centroides en los dos momentos de referencia.

Además, el usuario debe introducir las fechas que se corresponden con los dos momentos en que se ha evaluado la característica en cuestión. A partir de estos datos y de acuerdo con el valor del parámetro NIM (número de imágenes), que el usuario ha de definir, el programa establece NIM bloques diagrama, que representan la evolución de la distribución espacial de la característica entre los dos momentos de referencia temporal.

Para ello, el programa debe recurrir a calcular, en cada una de las NIM iteraciones, el valor de la característica sobre los centroides, antes de invocar a los procedimientos de construcción y dibujo de los diferentes bloques diagrama que explicitan el proceso de cambio.

Esta tarea la realiza una subrutina de interpolación temporal -INTEMP-, que se describe en el siguiente apartado.

El listado del programa, en cuya documentación se incluyen detalladamente los requisitos de uso, aparece a continuación. Los textos de todas las subrutinas que utiliza se encuentran en el apéndice final.

```

C *****
C * PROGRAMA DE DIBUJO DE BLOQUES DIAGRAMA EN PERSPECTIVA ISOMETRI-
C * CA, QUE EXPRESAN LA VARIACION DE UNA CARACTERISTICA EN EL INTERIOR
C * DE UN AREA, A PARTIR DE LAS COORDENADAS QUE DEFINEN SU CONTORNO Y
C * DE LOS VALORES DE LA CARACTERISTICA EN DIVERSOS PUNTOS INTERIORES
C * A ELLA, DE LOS CUALES SE NECESITA CONOCER TAMBIEN SU LOCALIZACION
C * ESPACIAL MEDIANTE COORDENADAS.
C * PARTIENDO DE LA LECTURA DE LOS VALORES DE LA CARACTERISTICA EN
C * DOS MOMENTOS TEMPORALES (T1 Y T2) Y DEL NUMERO DE IMAGENES A RE-
C * PRESENTAR (NIM), EL PROGRAMA DIBUJA NIM BLOQUES DIAGRAMA QUE EX-
C * Plicitan LA VARIACION TEMPORAL DE LA DISTRIBUCION ESPACIAL DE LA
C * CARACTERISTICA ENTRE EL MOMENTO T1 Y EL MOMENTO T2.
C * EL PROCEDIMIENTO DE INTERPOLACION TEMPORAL ES LINEAL.
C *
C * EL PROGRAMA PUEDE CONSIDERAR HASTA 300 CENTROIDES Y UN CONTORNO
C * DEFINIDO POR UN MAXIMO DE 100 PUNTOS. SI SE REBASAN ESTOS LIMITES,
C * ES NECESARIO REDIMENSIONAR LOS ARRAYS XV,YV,PV,D,B,X E Y.
C *****
C NOMBRE DEL PROGRAMA: FILM.
C LENGUAJE: FORTRAN77.
C AUTOR: JUAN ANTONIO CEBRIAN DE MIGUEL.
C DEPARTAMENTO DE GEOGRAFIA HUMANA.
C UNIVERSIDAD COMPLUTENSE (MADRID).
C JULIO 1982.
C ORDENADOR: FACOM M 180 N (INSTITUTO GEOGRAFICO NACIONAL).
C PLOTTER: CALCOMP 960 (DIBUJANDO SOBRE PLIEGOS DE 81(Y)*130(X) CMS).
C MEMORIA: 208 KBYTES.
C SUBROUTINAS NECESARIAS: INTEMP, BLOCKD, ZETA, RAY, REDRG, SETG2, G1,
C G2, VISTA, PLOT3, TRN32, CACHE, INTER, MERGE, INTP, PLOTMM, CNPEN.
C *****
C *****
C ORGANIZACION DE LOS DATOS PARA USO DEL PROGRAMA.
C LA INFORMACION SE ORGANIZA EN TRES FICHEROS.
C FICHERO FT05F001 (REUNE TODOS LOS PARAMETROS DE CONTROL).
C *****
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL. |.....|.....|.....|.....|.....|.....|.....|
C //FT05F001 DD *
C

```

C SEGUNDA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|.....|

AAAA BBBB CCCC.CCCC

C A = COORDENADA X DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLD  
 C QUE DIAGRAMA.  
 C B = COORDENADA Y DEL ORIGEN DE COORDENADAS DURANTE EL DIBUJO DEL BLD  
 C QUE DIAGRAMA.  
 C C = FACTOR DE ESCALA DEL BIBUJO.

C TERCERA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|.....|

DDDD EEEE FFFF GGGGG HHH.HHH III.III

C D = COORDENADA X DEL PUNTO DE VISTA ELEGIDO.  
 C E = COORDENADA Y DEL PUNTO DE VISTA ELEGIDO.  
 C F = COORDENADA Z DEL PUNTO DE VISTA ELEGIDO.  
 C G = ANGULO (EN EL SENTIDO DE LAS AGUJAS DEL RELOJ) QUE SE DESEA  
 C QUE FORME EL RAYADO CON LA VISUAL DIRIGIDA DESDE EL PUNTO DE  
 C VISTA AL ORIGEN DE COORDENADAS.  
 C H = VALOR Y DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGO-  
 C NO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.  
 C I = VALOR X DE LA RETICULA ORTOGONAL DE PUNTOS INTERIORES AL POLIGO-  
 C NO QUE SE REQUIERE PARA EL DIBUJO DEL DIAGRAMA.

C CUARTA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|.....|

JJJJ KKKK.KKKK LLLL.LLLL MMMM.MMMM NNNN.NNNN

C J = VALOR DE LA FUNCION FUERA DEL CONTORNO.  
 C K = VALOR MINIMO DE LA FUNCION DENTRO DEL CONTORNO.  
 C L = VALOR MAXIMO DE LA FUNCION DENTRO DEL CONTORNO.  
 C M = FACTOR DE CORRECCION DE LA FUNCION (\*).  
 C N = FACTOR DE CORRECCION DE LA FUNCION (+).

C QUINTA TARJETA.  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|.....|

DDDD PPPP @@@@ RRRR

C Q = FECHA DEL MOMENTO INICIAL.  
 C P = FECHA DEL MOMENTO FINAL.  
 C @ = NUMERO DE IMAGENES QUE SE DESEAN ENTRE AMBOS MOMENTOS.  
 C R = TAMANO DE LOS CARACTERES CON QUE SE REPRODUCIRA LA FECHA DE  
 C CADA IMAGEN.

C SEXTA TARJETA (COMO EJEMPLO).  
 C 1 10 20 30 40 50 70 80  
 COL. |.....|.....|.....|.....|.....|.....|.....|.....|

(2F6.3,2F10.2)

C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD B LOS  
 C VALORES DE LAS COORDENADAS DE LOS CENTROIDES Y EL VALOR DE LA CARAC-  
 C TERISTICA EN CADA CENTROIDE, EN LOS DOS MOMENTOS DE REFERENCIA.

```

C SEPTIMA TARJETA (COMO EJEMPLO).
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C (8F10.0)
C
C DESCRIPCION DEL FORMATO CON EL QUE SE VAN A LEER EN LA UNIDAD 9 LOS
C VALORES DE LAS COORDENADAS DE LOS PUNTOS QUE DEFINEN EL CONTORNO EX-
C TERIOR DEL AREA.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT08F001 (NUMERO DE CENTROIDES Y VALORES DE SUS COORDENADAS
C ***** Y DE LA CARACTERISTICA EN ELLOS, EN AMBOS MO-
C MENTOS).
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C //FT08F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C AAA
C
C A = NUMERO DE CENTROIDES.
C
C TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y, PESO DE
C LA CARACTERISTICA DE CADA CENTROIDE EN EL PRIMER Y EN EL SEGUNDO MO-
C MENTO -POR ESTE ORDEN-, CON EL FORMATO DESCRITO.
C
C ULTIMA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C /*
C
C FICHERO FT09F001 (NUMERO DE PUNTOS QUE FORMAN EL CONTORNO Y VALORES
C ***** DE SUS COORDENADAS).
C
C PRIMERA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C //FT09F001 DD *
C
C SEGUNDA TARJETA.
C 1 10 20 30 40 50 70 80
COL |.....|.....|.....|.....|.....|.....|.....|
C AAA
C
C A = NUMERO DE PUNTOS DEL CONTORNO.
C
C TERCERA TARJETA Y SIGUIENTES: COORDENADA X, COORDENADA Y DE CADA
C UNO DE LOS PUNTOS -POR ESTE ORDEN-, CON EL FORMATO DESCRITO.
C
C

```

```

C   ULTIMA TARJETA.
C   1      10      20      30      40      50      70      80
COL.1|.....|.....|.....|.....|.....|.....|.....|
C   /*
C   *****
C   *****
C   *****
C
C   DIMENSION PV(300,3)
C   CHARACTER FMT(80)
C   COMMON/TRID/XV(300),YV(300),B(2,100),XDR,YDR,FCT,VX,VY,VZ,WR,DR
C   *DS,ZB,ZO,Z1,ZF,ZINC,N
C
C   XV = VECTOR DONDE SE ALMACENAN LAS COORDENADAS X DE LOS CENTROIDES.
C   YV = VECTOR DONDE SE ALMACENAN LAS COORDENADAS Y DE LOS CENTROIDES.
C   PV(1,1) = VECTOR DONDE SE ALMACENAN LOS PESOS DE LA VARIABLE SOBRE
C   LOS CENTROIDES, EN EL MOMENTO INICIAL.
C   PV(1,2) = VECTOR DONDE SE ALMACENAN LOS PESOS DE LA VARIABLE SOBRE
C   LOS CENTROIDES, EN EL MOMENTO FINAL.
C   PV(1,3) = VECTOR DONDE SE ALMACENAN LOS PESOS DE LA VARIABLE SOBRE
C   LOS CENTROIDES, INTERPOLADOS EN CADA MOMENTO.
C   B = MATRIZ DONDE SE ALMACENAN LAS COORDENADAS QUE DESCRIBEN EL CON-
C   TORNO DEL AREA.
C   FMT = VECTOR DE CARACTERES PARA LA DESCRIPCION DEL FORMATO DE LOS DA-
C   TOS APORTADOS POR EL USUARIO.
C   *****
C
C   LECTURA DE DATOS.
C   *****
C
C   LECTURA DE PARAMETROS.
C
C   READ(5,201) XDR,YDR,FCT
C   WRITE(6,201) XDR,YDR,FCT
C 201 FORMAT(5X,2F5.0,F10.4)
C   READ(5,202) VX,VY,VZ,WR,DR,DS
C   WRITE(6,202) VX,VY,VZ,WR,DR,DS
C 202 FORMAT(5X,3F5.0,F6.0,2F8.3)
C   READ(5,203) ZB,ZO,Z1,ZF,ZINC
C   WRITE(6,203) ZB,ZO,Z1,ZF,ZINC
C 203 FORMAT(5X,F5.0,3F10.4,T40,F10.4)
C   READ(5,204) T1,T2,NIM,TAM
C   WRITE(6,204) T1,T2,NIM,TAM
C 204 FORMAT(5X,2F5.0,I5,F5.0)
C
C   LECTURA DE LAS COORDENADAS DE LOS CENTROIDES Y DEL PESO DE LA VA-
C   RIABLE EN ELLOS, EN LOS DOS MOMENTOS TEMPORALES DE REFERENCIA.
C
C   READ(8,100) N2
C   WRITE(6,100) N2
C 100 FORMAT(2X,I3)
C   READ(5,107) FMT
C 107 FORMAT(80A1)
C   WRITE(6,108) FMT
C 108 FORMAT('FORMATO DE LOS DATOS = ',80A1/)
C   READ(8,FMT) (XV(I),YV(I),PV(I,1),PV(I,2)),I=1,N2

```

```

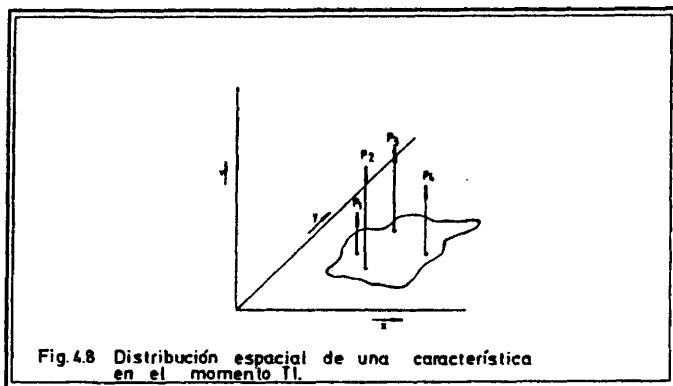
WRITE(6,103) (XV(I),YV(I),PV(I,1),PV(I,2),I=1,N2)
103 FORMAT(2X,4F12.4)
C
C LECTURA DE LA COORDENADAS DEL CONTORNO EXTERIOR.
C
READ(9,100) N
WRITE(6,100) N
READ(5,107) FMT
WRITE(6,104) FMT
104 FORMAT('OFORMATO DEL CONTORNO = ',80A1/)
READ(9,FMT) (B(1,I),B(2,I),I=1,N)
WRITE(6,105) N,(B(1,I),B(2,I),I=1,N)
105 FORMAT(13/(2F10.3))
C
C*****
C*****
C
C SE ABRE EL DIBUJO.
C
CALL PLOTS(0,0,6)
C
C SE LLAMA A LA RUTINA DE INTERPOLACION TEMPORAL, QUE, A SU VEZ,
C LLAMARA A BLOCKD PARA PINTAR LAS SUCEASIVAS IMAGENES.
C
CALL INTEMP(PV,300,N2,T1,T2,NIM,TAM)
C
C SE CIERRA EL DIBUJO.
C
CALL PLOT(-1.,-1.,999)
C
C*****
C*****
C
C STOP
C END

```

#### 4.3.2. El problema de la interpolación temporal.

Supongamos un caso sencillo del problema que se pretende resolver.

En la figura 4.8., aparecen un contorno y el peso de una característica en una serie de puntos interiores a él en el momento  $T_1$ . Con estos datos, los procedimientos de dibujo de bloques diagrama en perspectiva, que hemos definido en el capítulo 3, podrían empezar a funcionar y reproducir el volumen coherente con la situación en el momento  $T_1$ .



En la figura 4.9., se presentan los valores de la misma característica en el mismo ámbito, en el momento  $T_2$ .

Vamos ahora -Fig. 4.10.- a considerar los casos  $P_1$ ,  $P'_1$  y  $P_3$ ,  $P'_3$ .

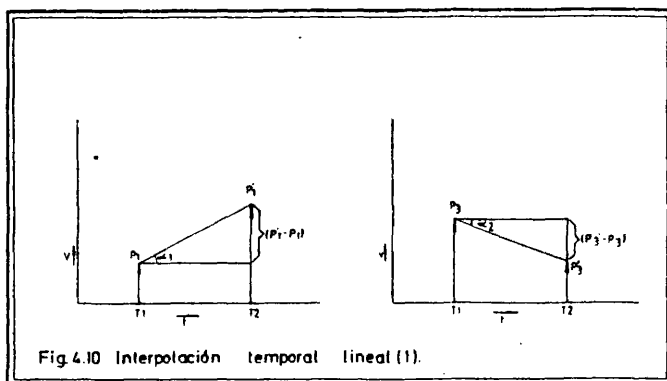
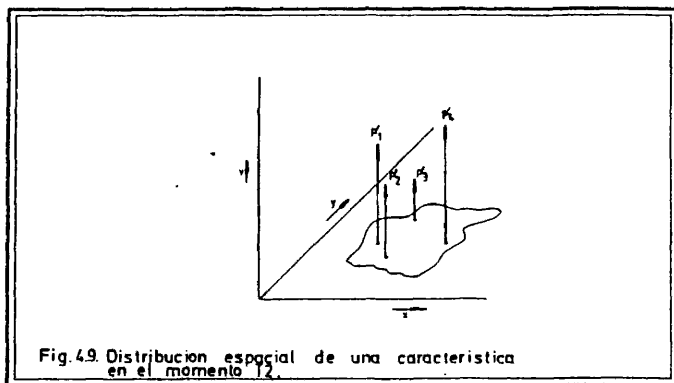
En ambos casos, la recta de interpolación que une los puntos que representan dos momentos temporales ( $T_1$  y  $T_2$ ) forma con la horizontal un ángulo ( $\alpha_1$  y  $\alpha_2$ , respectivamente) cuya magnitud puede conocerse, a partir de los valores de la característica en ambos momentos y del intervalo que los separa.

$$/4.1/ \alpha_1 = \arctg (P'_1 - P_1) / (T_2 - T_1)$$

$$/4.2/ \alpha_2 = \arctg (P'_3 - P_3) / (T_2 - T_1)$$

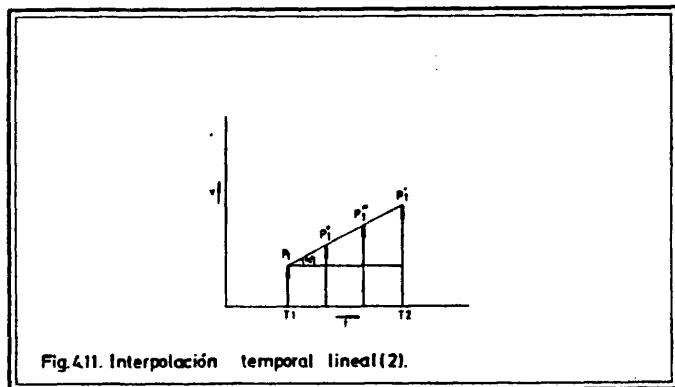
Todos estos datos son conocidos por el programa, porque han sido introducidos por el usuario. Vamos a ver a continuación cómo





puede establecerse el peso de la característica sobre los centros ides en los momentos intermedios entre T1 y T2.

Supongamos que el usuario ha decidido que se establezcan dos imágenes intermedias entre los momentos T1 y T2 (Fig. 4.11.).



El programa tendrá que calcular, en función de los datos que conoce  $-T1, P = \{P_i, i=1, n\}, T2, P' = \{P'_i, i=1, n\}$  -, el valor de  $P'' = \{P''_i, i=1, n\}$  y pintar el bloque diagrama coherente con esta lista, y el valor de  $P''' = \{P'''_i, i=1, n\}$  y efectuar, a continuación, la misma operación de dibujo.

La figura 4.12. es una ampliación de la figura 4.11.

En ella se cumple que:

$$/4.3./ \quad \overline{AH} = P_1$$

$$/4.4./ \quad \overline{GI} = P'_1$$

$$/4.5./ \quad \overline{GF} = (P'_1 - P_1)$$

$$/4.6./ \quad \overline{AF} = (T2 - T1)$$

$$/4.7./ \quad \operatorname{tg} \alpha_1 = GF/AF$$

$$/4.8./ \quad P''_1 = P_1 + CB$$

$$/4.9./ \quad P'''_1 = P_1 + ED$$

$$/4.10./ \quad AB = BD = AF/(NIM+1).$$

NIM= n° de imágenes intermedias a reproducir.

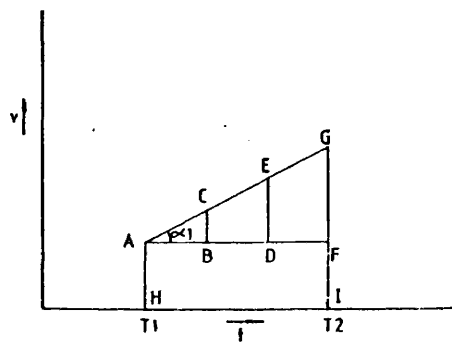


Fig.4.12. Interpolación temporal lineal (3).

Las dos únicas incógnitas de toda esta serie de ecuaciones son CB y ED. Una vez deducidos sus valores podremos conocer los pesos  $P_1''$  y  $P_1'''$  en función de /4.8/ y /4.9/.

En el triángulo  $\hat{ABC}$  se cumple que:

$$/4.11/ \quad CB/AB = \operatorname{tg} \alpha_1, \text{ luego } /4.12/ \quad CB = AB \cdot \operatorname{tg} \alpha_1.$$

En el triángulo  $\hat{ADE}$ ,

$$/4.13/ \quad ED/(AB+BD) = \operatorname{tg} \alpha_1, \text{ luego } /4.14/ \quad ED = (AB+BD) \cdot \operatorname{tg} \alpha_1$$

En /4.12/ y /4.14/ se puede comprobar que CB y ED son función, también, de los datos que introduce el usuario.

Visto el procedimiento, sencillo, de derivar los pesos de la variable en las NIM situaciones intermedias entre los momentos T1 y T2, la automatización de los cálculos se puede consultar en el listado de la subrutina INTEMP, que aparece en el apéndice final.

La subrutina BLOCKD, que utiliza INTEMP para dibujar cada una de las situaciones intermedias, es una versión, en forma de subrutina, del programa DIABLOCK, que hemos descrito en el capítulo anterior.

#### 4.3.3. Filmación de los resultados.

Para probar las prestaciones de este tipo de tratamientos en análisis de procesos espacio-temporales, hemos realizado una secuencia animada que representa la evolución de las densidades provinciales de población en la España peninsular entre 1900 y 1980. La filmación se ha realizado en video y su reproducción, por tanto, se obtiene sobre una pantalla de televisión.

Partiendo de los datos decenales de las densidades provinciales de población -en el periodo 1900-1980- y de la digitización del contorno de la España peninsular y de las localizaciones de las capitales de provincia (1), se ha utilizado el programa FILM para obtener ochenta y un dibujos, cada uno de los cuales representa la distribución, real o interpolada, de la densidad de población en cada año del periodo. Para ello han sido necesarias ocho pasadas del programa.

Los dibujos se han reproducido en papel, utilizando el plotter CALCOMP 960 del Instituto Geográfico Nacional.

Posteriormente, se han grabado los dibujos, a razón de 12 imágenes por cada uno de ellos, con una cámara de video (2), habiéndose registrado también las fechas correspondientes a cada gráfico, para obtener una escala temporal de interpretación de la secuencia animada.

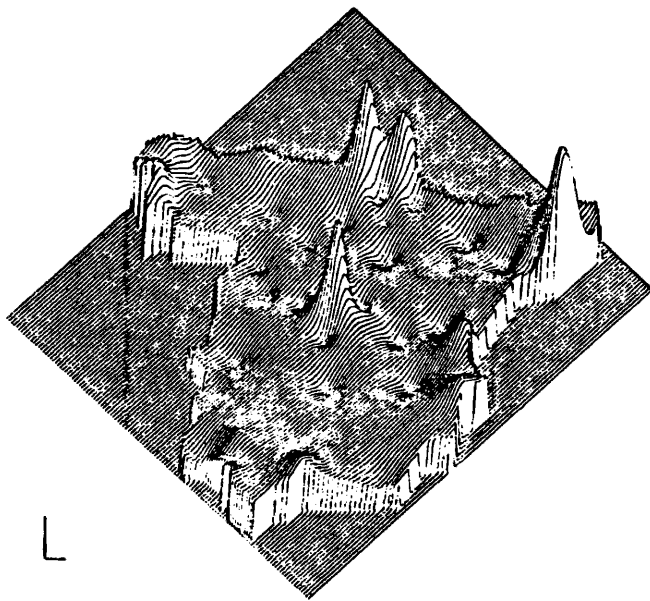
El resultado es una animación gráfica, elocuente, de 40 segundos de duración, que constituye una muestra importante de las prestaciones de este tipo de procedimientos en análisis de procesos de cambio espacio-temporal.

Cuando, por la disposición de las máquinas adecuadas, se pueda obtener la automatización del grabado de los fotogramas de base, se habrá dado un paso importante en la viabilidad efectiva de esta clase de tratamientos en trabajos de investigación geográfica en España.

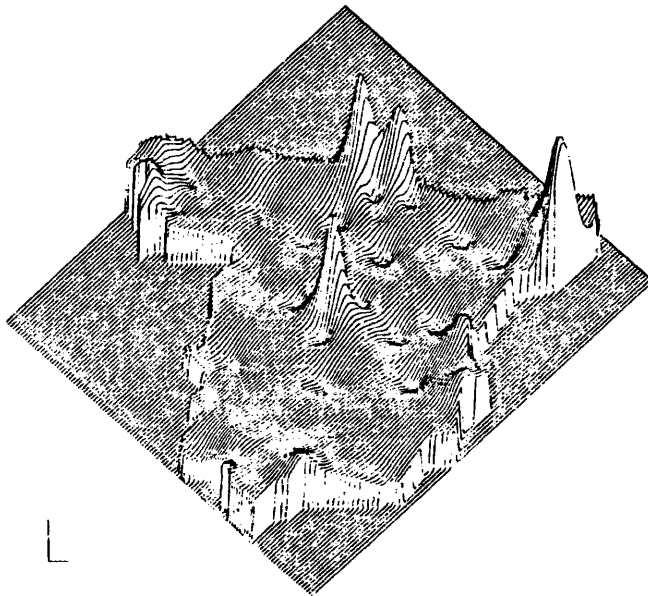
Presentamos, finalmente, una serie de once de los dibujos que han sido grabados en video para simular la animación. Corresponden al periodo 1950-1960.

Los bloques diagrama de los años 1950 y 1960 se han efectuado en base a los datos disponibles, los restantes representan valores interpolados.

Evolución de las densidades  
de población provinciales en el  
periodo 1950-1960.

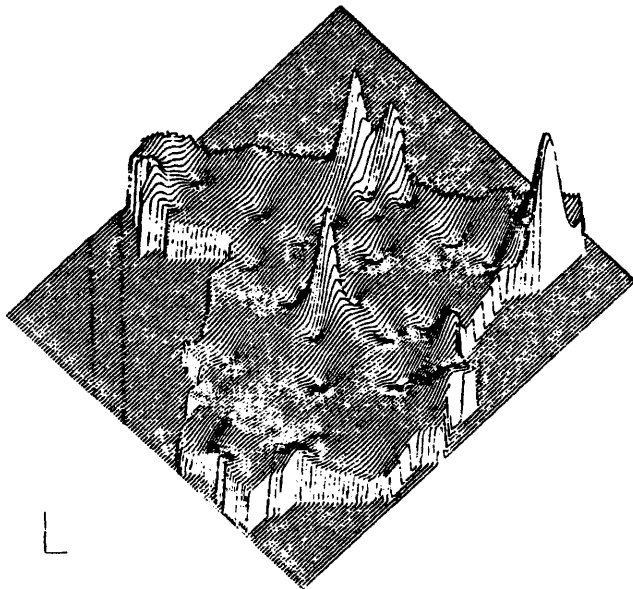


1950



L

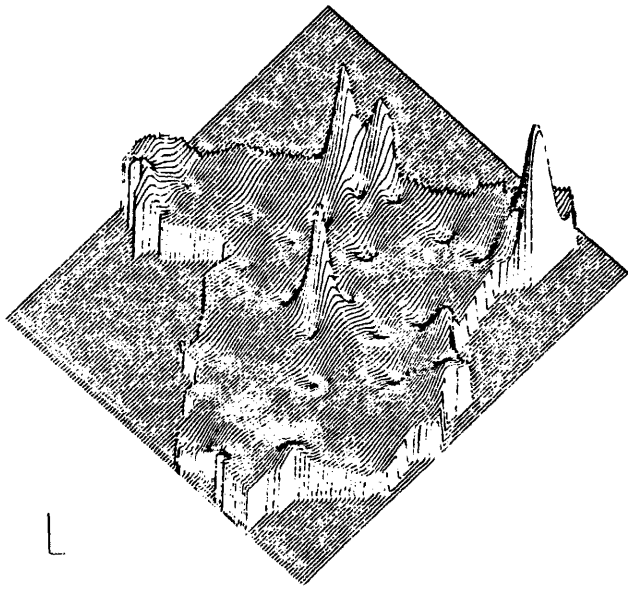
1951



L

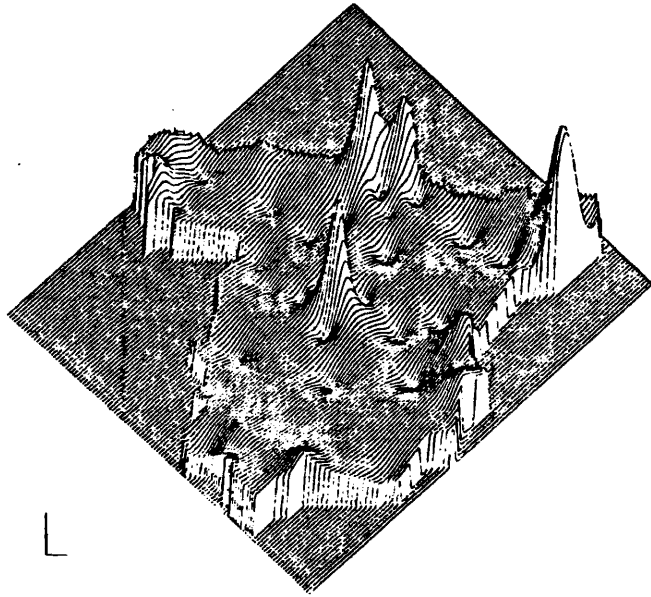
1952



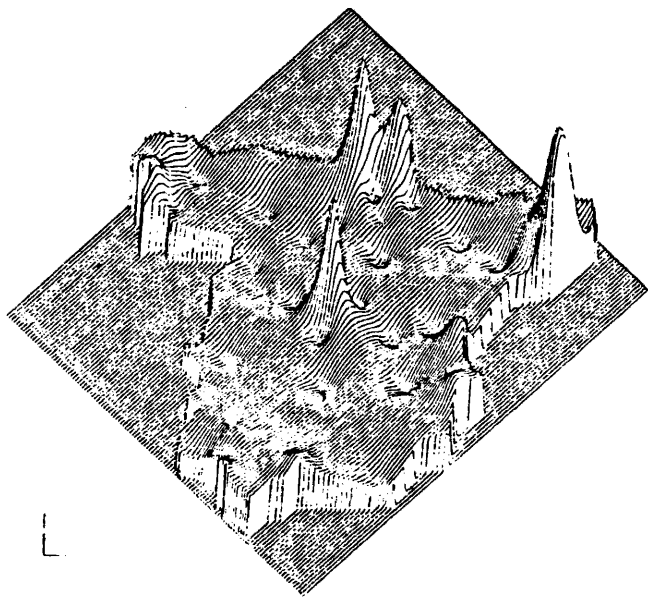


L

1955

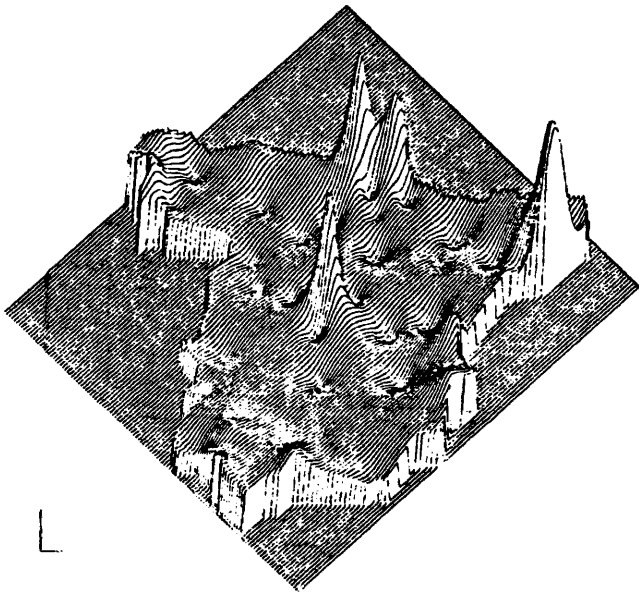


1954

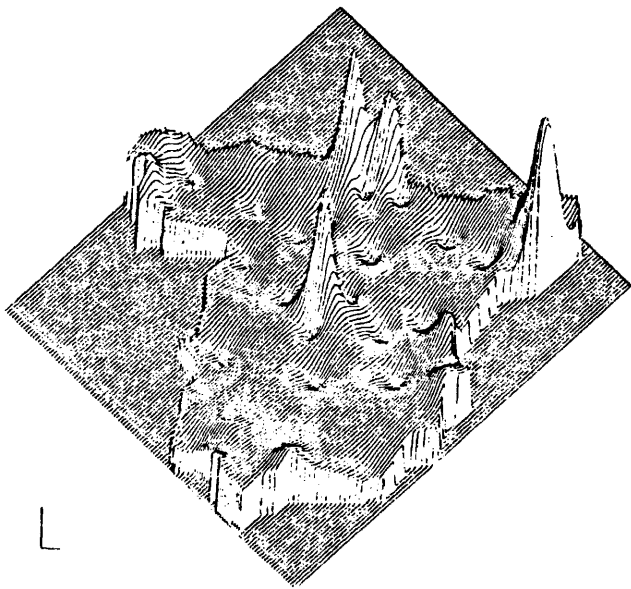


L

1955

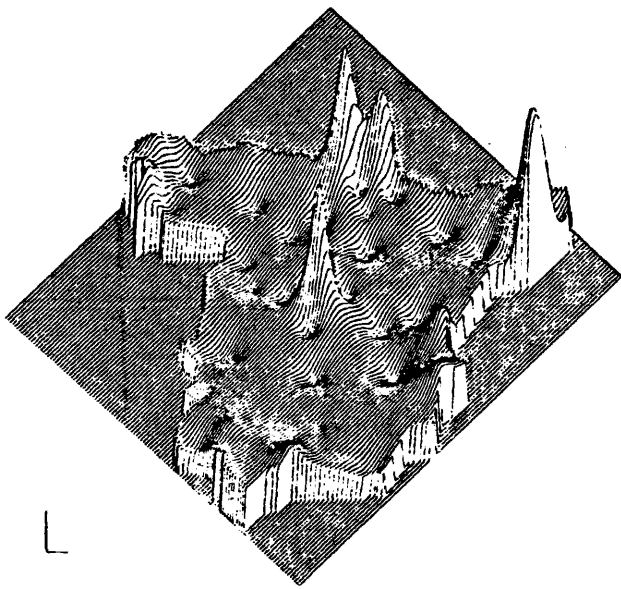


1356

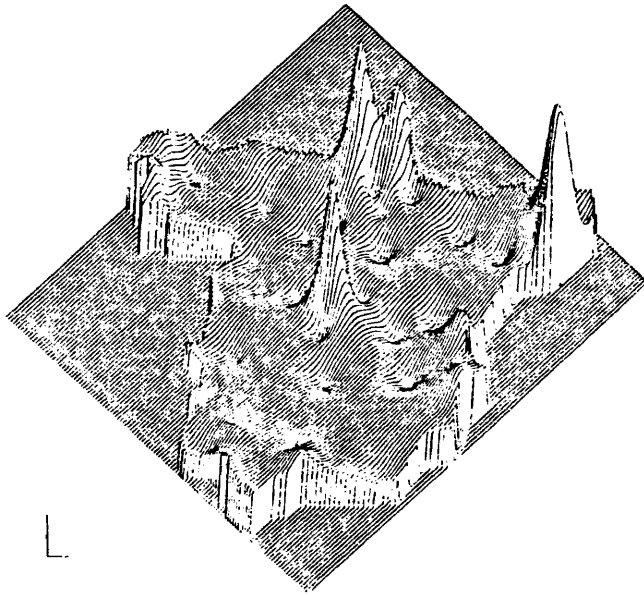


L

1957



1958



L

1953



L

1960

Fig.4.13.



NOTAS AL CAPITULO 4.

- (1) Todos estos datos me los ha proporcionado grabados en un so porte informático Beatriz Jiménez Blasco.
- (2) A pesar de haber tomado muchas precauciones durante la graba ción, para que no se alteraran los ejes de referencia básica de todos los dibujos, ha sido imposible evitar que se produjeran ciertas oscilaciones. Si se hubiera podido disponer de al gún procedimiento de grabación automática de fotogramas, no existirían estas anomalías.

263

CONCLUSIONES

En las conclusiones de esta tesis, junto a una síntesis de los resultados más importantes de ésta, vamos a señalar con determinimiento qué nuevas vías de investigación y trabajo es necesario acometer en lo sucesivo, en el marco de la Geografía española, en el campo del tratamiento digital de imágenes. Nuevas aplicaciones que, tomando como punto de partida las realizaciones que hemos descrito a lo largo de los capítulos precedentes, completen el amplio espectro de posibilidades que la tecnología informática, en continua evolución, ofrece al proceso de datos en ciencias espaciales.

Esto es así porque la primera conclusión que se puede establecer acerca de la presente tesis es que su principal característica la constituye el hecho de haber iniciado un nuevo ámbito de trabajo en la investigación geográfica española. Esta tesis abre muchas puertas y no cierra ninguna. Su principal virtud, eso creemos, es su carácter innovador y su planteamiento abierto a nuevas realizaciones.

Se configura, por tanto, un nuevo campo de investigación que no puede ser ignorado, si se pretende ofrecer a la Geografía española un sistema integrado y completo de tratamiento digital de información espacial.

Los trabajos que han de acometerse en los próximos años están ya definidos en parte y, algunos, en vías de realización. Vamos ahora, por tanto, a describir el panorama de lo que pensamos ha de ser la prolongación de la presente tesis, en un plazo relativamente corto.

En primer lugar, y ciñéndonos a los temas que han sido considerados en cada uno de los capítulos que anteceden, hay que señalar que ya están en marcha trabajos de ampliación de las posibilidades de representación gráfica y cartográfica automatizada.

Se continúa trabajando en la confección de programas de diseño automático que completan la gama de diagramas de uso frecuente en investigación geográfica.

En el campo de la representación cartográfica en dos dimensiones está definido el proyecto de creación de un programa ambicioso de cartografía temática que contemple un variadísimo espectro de signos convencionales, sin olvidar aquellos que explicitan flujos de intercambio espacial. También se pretende disponer, en breve, de subrutinas de dibujo automático de isolíneas.

Para completar las posibilidades de representación de volúmenes en perspectiva, se están desarrollando programas de represen

tación de sólidos por aristas y de superficies por isolíneas en tres dimensiones.

Con vistas a disponer de procedimientos eficaces de cartografía dinámica, se pretende llegar al establecimiento de sistemas de cartografía dinámica en tiempo real, accediendo a los periféricos más adecuados a este tipo de tratamientos.

Para hacer viables estos y otros proyectos será necesario, junto al mantenimiento y desarrollo de la tecnología vectorial de descripción de imágenes, recurrir a las técnicas raster, de representación en forma matricial.

En segundo término, vemos necesario acometer en profundidad las técnicas de análisis estadístico de distribuciones espaciales, ya que se trata de un desarrollo coherente con el esfuerzo de descripción numérica de la configuración ambiental que ha sido, y continuará siendo, realizado en los trabajos de cartografía automática.

Finalmente, consideramos que no se debe eludir el estudio detallado de las diversas estrategias de definición e implementación de base de datos georeferenciados. Es ésta una tarea básica y fundamental que condiciona cualquier proyecto viable de cartografía automática y de análisis estadístico de las estructuras espaciales. Entre otros temas, resulta de especial interés el estudio de las posibilidades de implementación de este tipo de bases de datos a partir de la información grabada digitalmente por sensores remotos (scanners multiespectrales).

Dos razones fundamentales justifican que todo este conjunto de nuevas tareas que se sugieren para un futuro próximo no hayan sido incluidas en el presente trabajo. La primera de ellas es que se encuentran más allá de sus objetivos iniciales. En esta tesis no se ha pretendido nunca agotar el campo de las realizaciones de representación gráfica y cartográfica automatizadas. Su objetivo ha sido siempre el de sentar unas bases sólidas para posteriores desarrollos, presentado un conjunto de aplicaciones, con entidad por sí mismas, eso sí, que demuestran las enormes posibilidades del procesado automático de datos espaciales.

El segundo motivo lo constituye la complejidad de todos estos procedimientos y la gran dedicación en tiempo y esfuerzo que requiere su resolución. La ampliación de los trabajos que se presentan en esta tesis no es una tarea trivial. Tener resueltos todos los problemas que apunto como continuación del trabajo de dos años y medio supondría situarse en la vanguardia mundial en este campo de investigación, y no podemos olvidar que han sido necesarios más de veinte años de trabajo -desde las primeras sugerencias de Tobler /165/ en 1959- para llegar a los planteamientos actuales.

Centrándonos ya en las aportaciones propias de esta tesis, es oportuno señalar, respecto a todo su conjunto, que se puede ha

blar ya de un software autóctono -desarrollado por la Geografía española- de aplicaciones de tratamiento gráfico y cartográfico automático. Este hecho no deja de tener su importancia, pues a partir de ahora, si bien se puede seguir recurriendo a la adquisición selectiva de programas desarrollados en el extranjero, muchas de las necesidades de representación gráfica y cartográfica habituales en la investigación geográfica quedan perfectamente cubiertas por los procedimientos que se han desarrollado a lo largo de esta tesis. Además, en base a las capacidades adquiridas, se dispone de los elementos de juicio necesarios para hacer una crítica más adecuada de los procedimientos extranjeros, y para ampliarlos o modificarlos en función de los objetivos propios.

Como resultado de esta tesis, en primer lugar, se ha conseguido liberar a la investigación geográfica española de las tareas de tratamiento gráfico manual de series estadísticas. No están contempladas todas las posibilidades, pero el conjunto de programas presentados en el primer capítulo, unido al de los que realizó en su momento Martínez de Sola /105/, tiene ya una cierta entidad. Además, se dispone de todos los medios para hacer frente a cuantas nuevas peticiones se puedan formular en este sentido.

También ha quedado resuelto de forma automatizada el establecimiento de mapas de coropletas, de muy frecuente uso y de costosa confección, cuando se pretende obtener una representación de calidad.

Hasta ahora, sólo hemos considerado tareas que pueden realizarse fácilmente a mano, aunque con un despilfarro de tiempo considerable. Pero, además, en esta tesis se han desarrollado, y se ofrecen como herramientas de trabajo que no requieren una especial pericia para su utilización, procedimientos de obtención de gráficos irrealizables -con rapidez, flexibilidad y precisión- manualmente. Me refiero a los programas de obtención automática de bloques diagrama en perspectiva isométrica, en sus diversas modalidades.

Por último, se ha podido resolver, aunque de forma provisional -en espera de acceder a los periféricos gráficos adecuados para este tipo de tratamientos-, la obtención automática de secuencias animadas que representan deformaciones de volúmenes en función del tiempo. Estos procedimientos constituyen sin duda, un instrumento analítico de primer orden en el estudio de procesos espacio-temporales.

Pero antes de terminar, quiero dejar constancia clara de que será muy difícil atender a todas estas demandas sin disponer del conjunto de máquinas necesarias. Si no se arbitran los medios para dotar a los Departamentos de Geografía españoles del hardware imprescindible para trabajos de esta índole, se perderá la posibilidad de que se desarrollen realmente nuevas técnicas y, sobre todo -lo que no es menos importante-, se imposibilitará la transmisión adecuada de este tipo de conocimientos a las nuevas generaciones de alumnos.

INDICE DE SUBROUTINAS

	<u>Págs.</u>
<u>SUBROUTINAS:</u>	
AFINE . . . . .	270
AFIN2 . . . . .	271
AGREGA . . . . .	272
ANNAC . . . . .	273
AVER . . . . .	274
BLOCKD . . . . .	275
BLOCKDATA . . . . .	279
CACHE . . . . .	280
CHECK2 . . . . .	285
CLASCT . . . . .	286
CLASDT . . . . .	288
CLASST . . . . .	290
CLASUM . . . . .	292
CLOCK . . . . .	293
CLUSTR . . . . .	294
CNPEN . . . . .	298
CONCE2 . . . . .	300
CONTIG . . . . .	302
COROPL . . . . .	305
CUADRA . . . . .	307
CUADRA . . . . .	308
DEPURA . . . . .	309
DIAGRM . . . . .	310
DIBUJO . . . . .	313
DIFER. . . . .	318
G1 . . . . .	319
G2 . . . . .	320
INPUT . . . . .	321
INTEMP . . . . .	325
INTER . . . . .	327
INTERS . . . . .	328
INTP . . . . .	329
INVERT . . . . .	330
LEGEND . . . . .	331
LEYEN1 . . . . .	334

<u>SUBROUTINAS :</u>	<u>Págs.</u>
LEYEN2 . . . . .	335
MARCO . . . . .	336
MERGE . . . . .	337
MOTA . . . . .	339
NEWAI . . . . .	340
NEWWAY . . . . .	341
ORDEN . . . . .	343
OTHWAY . . . . .	344
PIRAM . . . . .	345
PIRAM2 . . . . .	348
PLOT3 . . . . .	350
PLOTMM . . . . .	351
PLOTM2 . . . . .	352
POLIG . . . . .	353
POLIG2 . . . . .	355
POLIG3 . . . . .	357
PORCEN . . . . .	359
PORCE2 . . . . .	360
RAY . . . . .	361
RAY . . . . .	363
REINSC . . . . .	366
REORG . . . . .	367
ROTULO . . . . .	368
SCVERT . . . . .	369
SETG2 . . . . .	371
TEST . . . . .	372
TESTFA . . . . .	373
TESTFE . . . . .	374
TITULO . . . . .	375
TRAMA . . . . .	377
TRAMA . . . . .	378
TRAMEL . . . . .	379
TRAMEL . . . . .	380
TRAMLY . . . . .	381

<u>SUBROUTINAS</u>	<u>Págs.</u>
TRANFR . . . . .	383
TRANF2 . . . . .	384
TRANSF . . . . .	385
TRANS1 . . . . .	387
TRANS2 . . . . .	388
TRIANG . . . . .	389
TRN32 . . . . .	393
VISTA . . . . .	394
ZETA . . . . .	396
ZETA2 . . . . .	397



```

C
C *****
C * ESTA SUBROUTINA SIMPLIFICA EL DIBUJO DE LA RAYA TRIDIMENSIONAL *
C * EN LOS PUNTOS PERTENECIENTES A FRONTERAS COMUNES A DOS AREAS. *
C *****
C
SUBROUTINE AFINE(LINEAA, I32, I49)
REAL LINEAA(3, I32)
I=1
J=0
I32M1=I32-1
11 CONTINUE
IF(((LINEAA(1, J+1)-LINEAA(1, I)).LT.0.00001).AND.(ABS(LINEAA(3, J+
+1)-LINEAA(3, I)).LT.0.00001)) GOTO 3
J=J+1
DO 2 K=1, 3
2 LINEAA(K, J)=LINEAA(K, I)
GOTO 1
3 I=I+1
1 I=I+1
IF(I.LE. I32M1) GOTO 11
J=J+1
DO 4 K=1, 3
4 LINEAA(K, J)=LINEAA(K, I32)
I49=J
RETURN
C
C *****
C
END

```

```
C
C *****
C *   ESTA SUBROUTINA DEPURA LOS POSIBLES ERRORES DE 'SOLAPAMIENTO' *
C *QUE SE HAN PODIDO COMETER EN LA DIGITIZACION DE LOS CONTORNOS. *
C *****
C
SUBROUTINE AFIN2(A,N,INDIC)
DIMENSION A(N,3)
INDIC1=INDIC-1
DO 1 I=1,INDIC1
IF(A(I,2).LE.A(I+1,1))GOTO 1
B=(A(I,2)+A(I+1,1))/2.
A(I,2)=B
A(I+1,1)=B
1 CONTINUE
RETURN
C
C *****
C
END
```

```

C
C *****
C * SUBROUTINE DE AGREGACION DE POBLACIONES CLASIFICADAS POR EDAD Y *
C * SEXO. *
C *****
C
C SUBROUTINE AGREGA(PM,PF,N,L,IEND,IFIRST,NN)
C
C PM Y PF = VECTORES EN LOS QUE ESTAN ALMACENADOS LOS DATOS DE ES-
C TRUCTURA POR EDAD Y SEXO, A AGREGAR.
C N = NUMERO DE GRUPOS DE EDAD ANTES DE LA AGREGACION.
C L = LONGITUD DE LOS VECTORES PM Y PF EN EL PROGRAMA QUE LLA-
C MA A ESTA SUBROUTINA.
C IEND = AMPLITUD EN AÑOS DE LOS GRUPOS DE EDAD DESPUES DE LA AGRE-
C GACION.
C IFIRST = AMPLITUD EN AÑOS DE LOS GRUPOS DE EDAD ANTES DE LA AGREGA-
C CION.
C NN = NUMERO DE GRUPOS DE EDAD DESPUES DE LA AGREGACION.
C *****
C
C DIMENSION PM(L),PF(L)
C *****
C
C NO=IEND/IFIRST
C NN=0
C DO 1 I=1,N,NO
C NN=NN+1
C SUM=0.
C SUMF=0.
C DO 2 J=1,L
C IJ=I+J-1
C SUM=SUM+PM(IJ)
C SUMF=SUMF+PF(IJ)
C 2 CONTINUE
C PM(NN)=SUM
C PF(NN)=SUMF
C 1 CONTINUE
C *****
C
C RETURN
C END

```

```

C
C *****
C *   ESTA SUBROUTINA SITUA, A LA DERECHA DE LA PIRAMIDE, LA LEYENDA *
C *DE LOS AÑOS DE NACIMIENTO DE LAS DIVERSAS COMORTES. *
C *****
C
SUBROUTINE ANNAC(J,AN)
COMMON/A/DATOS(90,40),ICOD(20),NGE(20),IAG(20),IAGF(20)
COMMON/D/ESCAL,SEPAR,ESCA,SEPAR1,BASE,SEPAR2,SEPAR3
CALL PLOT((2*BASE+SEPAR1+SEPAR),SEPAR2,-3)
DX=0.2
TAM=(SEPAR1-0.2)/5.
K=NGE(J)+1
DO 1 I=1,K
CALL PLOT(DX,ESCA*(I-1),2)
CALL NUMBER(DX,ESCA*(I-1),TAM,(AN-IAGF(J))*(I-1),0.,-1)
IF(I.EQ.K) GOTO1
CALL PLOT(0.,ESCA*(I-1),3)
CALL PLOT(0.,ESCA*I,2)
1 CONTINUE
CALL PLOT(-(2*BASE+SEPAR1+SEPAR),-SEPAR2,-3)
C
C *****
C
RETURN
END

```

```

C
C *****
C * SUBPROGRAMA FUNCTION CUYA ENTRADA AVER CALCULA LA MEDIA ARIT- *
C *METICA DE N VALDRES Y CUYA ENTRADA DESTIP CALCULA SU DESVIACION *
C *STANDARD. *
C *****
C
FUNCTION AVER(N)
COMMON/A/P(100)
T=0.
DO 1 I=1,N
T=T+P(I)
1 CONTINUE
AVER=T/FLOAT(N)
C*****
RETURN
C*****
ENTRY DESTIP(N)
DC=0.
DO 2 I=1,N
DC=DC+(P(I)-AVER)*(P(I)-AVER)
2 CONTINUE
DESTIP=SQRT(DC/FLOAT(N))
RETURN
C
C*****
C
END

```

```

C
C *****
C * SUBROUTINA DE DIBUJO DE BLOQUES DIAGRAMA EN PERSPECTIVA ISOMETRI*
C *CA, QUE EXPRESAN LA VARIACION DE UNA CARACTERISTICA EN EL INTERIOR*
C *DE UN AREA, A PARTIR DE LAS COORDENADAS QUE DEFINEN SU CONTORNO Y *
C *DE LOS VALORES DE LA CARACTERISTICA EN DIVERSOS PUNTOS INTERIORES *
C *A ELLA, DE LOS CUALES SE NECESITA CONOCER TAMBIEN SU LOCALIZACION *
C *ESPACIAL MEDIANTE COORDENADAS.
C *****
C
C SUBROUTINE BLOCKD(PV,MAX,N2,DATE,TAM)
C
C PV = PESOS DE LA VARIABLE SOBRE LOS CENTROIDES.
C MAX = DIMENSION DE PV EN EL PROGRAMA PRINCIPAL.
C N2 = NUMERO DE CENTROIDES.
C DATE = MOMENTO TEMPORAL AL QUE CORRESPONDE EL DIBUJO DEL DIAGRAMA.
C TAM = TAMAÑO DE LOS CARACTERES CON QUE SE REPRODUCE ESTE.
C *****
C
C COMMON/TRID/XV(300),YV(300),B(2,100),XOR,YOR,FCT,VX,VY,VZ,WR,DR
C *,DS,ZB,ZO,ZI,ZF,ZINC,N
C COMMON/XYMM/XH2,YM2,XM1,YM1,FCT1
C DIMENSION PV(MAX),D(300)
C DIMENSION A(30)
C DIMENSION X(100),Y(100)
C DIMENSION W(3,600),V(2,600)
C REAL LINEA(2,300),LINEAA(3,300)
C LOGICAL LW(600)
C
C PARA LA DESCRIPCIÓN DE LA MEMORIA, VER DIABLOCK.
C *****
C
C INICIALIZACION DE VARIABLES DE CONTROL DE ESCRITURA.
C
C DATA IDBS,NDIS/0,1/
C
C INICIALIZACION DE VALORES DE LA ZONA COMUN XYMM PARA AVERIGUAR EL
C RECUADRO.
C
C XM1= 1.E74
C XM2=-1.E74
C YM1= 1.E74
C YM2=-1.E74
C FCT1=FCT
C *****
C
C SE CONSIDERA EL FACTOR DE ESCALA.
C
C CALL FACTOR(FCT)
C
C SE DESPLAZA EL ORIGEN DE COORDENADAS.
C
C CALL PLOT(XOR,YOR,-3)
C

```

```

C SE CALCULA LA MATRIZ DE TRANSFORMACION DE COORDENADAS EN EL ESPACIO
C DE TRES DIMENSIONES, EN COORDENADAS EN EL ESPACIO BIDIMENSIONAL DE
C DIBUJO.
C
C CALL VISTA(VX,VY,VZ)
C
C SE PINTA UN TRIEDRO EN EL ORIGEN DE COORDENADAS.
C
C CALL PLOT3(1.,0.,0.,3)
C CALL PLOT3(0.,0.,0.,2)
C CALL PLOT3(0.,2.,0.,2)
C CALL PLOT3(0.,0.,3.,3)
C CALL PLOT3(0.,0.,0.,2)
C
C SE DEFINE EL ANGULO DE RAYADO COHERENTE CON EL PUNTO DE VISTA ELE-
C GIDO Y CON LA PETICION MANIFESTADA POR EL USUARIO.
C
C CALL SETG2(VX,VY,WRA)
C WRA=-WRA
C WR=WR-90.
C WR=WR+WRA
C
C SE GIRA EL POLIGONO, SE ENMARCA Y SE CALCULA EL NUMERO DE RAYAS NE-
C CESARIAS PARA REPRODUCIR EL BLOQUE DIAGRAMA, EN FUNCION DE LA RETI-
C CULA OPTOGONAL DEFINIDA POR EL USUARIO AL DAR VALORES A DR Y A DS.
C
C XMIN=1.E74
C XMAX=-1.E74
C YMAX=-1.E74
C YMIN=1.E74
C DO 1 I=1,N
C CALL G1(X(I),Y(I),B(1,I),B(2,I),WR)
C IF(YMAX.LT.Y(I)) YMAX=Y(I)
C IF(YMIN.GT.Y(I)) YMIN=Y(I)
C IF(XMAX.LT.X(I)) XMAX=X(I)
C IF(XMIN.GT.X(I)) XMIN=X(I)
C 1 CONTINUE
C WRITE(6,106) YMIN,YMAX,XMIN,XMAX
C WRITE(6,106) (X(I),Y(I),I=1,N)
106 FORMAT(/(10X,2F10.3))
C M=(YMAX-YMIN)/DR+1.5
C YR=YMIN-(DR+0.06789)
C
C*****
C M3M=0
C MAX=600
C*****
C
C EN ESTA ITERACION SE DIBUJA EL BLOQUE DIAGRAMA.
C
C*****
C DO 20 J=1,M
C*****
C
C 20-1. SE RAYA EL POLIGONO EN TRES DIMENSIONES.
C
C YR=YR+DR
C CALL RAY(X,Y,N,YR,A,K)

```

```

IF(K.LE.0) GO TO 20
I=0
DX=DS
XR=XMIN-DX
A(K+1)=999999.
ICONT=0
2 CONTINUE
XR=XR+DX
IF(XR.GT.XMAX+DX) GO TO 5
IF(XR.LE.A(I+1)) GO TO 3
I=I+1
3 CONTINUE
CALL G2(XN,YN,XR,YR,WR)
Z=ZB
IF(MOD(I,2).EQ.0) GO TO 4
C
C 20.1.1. LLAMADA A LA SUBROUTINA DE INTERPOLACION.
C
CALL ZETA(XN,YN,P,XV,YV,PV,D,N2)
IF(IDBS.GE.NDIS)
*WRITE(6,999) XR,YR,XN,YN,P
999 FORMAT(60X,5F10.3)
Z=ZF+(P-ZINC)
Z=AMAX1(Z,Z)
Z=AMIN1(Z,Z)
4 CONTINUE
C
C 20.2. SE CALCULAN LAS COORDENADAS DE LA RAYA EN EL ESPACIO IMAGEN.
C
ICONT=ICONT+1
LINEAA(1,ICONT)=XN
LINEAA(2,ICONT)=YN
LINEAA(3,ICONT)=Z
CALL TRNS2(LINEA(1,ICONT),LINEAA(1,ICONT))
GO TO 2
5 CONTINUE
C
C 20.3. SE CALCULAN LAS SECCIONES NO OCULTADAS POR SECCIONES ANTERIO-
C RES, Y SE DIBUJAN.
C
IER=0
CALL CACHE(V,M3M,M33,MAX,LINEA,ICONT,W,LW,0,IER,0)
M3M=M33
C
IF(IER.GE.6) STOP
C
C*****
20 CONTINUE
C*****
C*****
C
C SE SITUA LA FECHA A LA QUE CORRESPONDE LA IMAGEN.
C
CALL NUMBER((XM2/FCT+1.),YM1/FCT,TAM,DATE,0.,-1)
CALL WHERE(XDATE,YDATE,RFACT)
XM2= XDATE*FCT
C

```



```
C CONTROL DE LA PLUMA PARA SITUARLA ADECUADAMENTE EN EL SIGUIENTE DI-  
C BUJO.  
C  
C CALL PLOT(-XDR,-YDR,-3)  
C XM2=XM2+XDR*FCT  
C YM2=YM2+YDR*FCT  
C CALL CNPEN(XM2,YM2)  
C  
C SE DEVUELVE A WR EL VALDR CON EL QUE ENTRA A BLOCKD.  
C  
C WR=WR+90.-WRA  
C RETURN  
C  
C*****  
C  
C END
```



```

C
C *****
C *   ESTA SUBROUTINA RECIBE LA NUEVA LINEA A PINTAR Y LA ENVOLVENTE *
C *HASTA ESE MOMENTO. PINTA LAS SECCIONES DE LA NUEVA LINEA, NO OCUL- *
C *TADAS POR LA ENVOLVENTE Y CONSTRUYE LA NUEVA ENVOLVENTE. *
C *****
C
C   SUBROUTINE CACHE(V,M,MP,MAX,C,N,W,LW,ITIP,IER,IBDIS)
C
C   V = ENVOLVENTE.
C   M = LONGITUD DE LA ENVOLVENTE A LA ENTRADA A ESTA SUBROUTINA.
C   MP = LONGITUD DE LA NUEVA ENVOLVENTE.
C   MAX = DIMENSION MAXIMA DE V,C,W Y LW EN EL PROGRAMA PRINCIPAL.
C   C = NUEVA LINEA A PINTAR.
C   N = LONGITUD DE LA NUEVA LINEA A PINTAR.
C   W = ESPACIO DE TRABAJO PARA REALIZAR EL FUNDIDO DE LA NUEVA LINEA
C   CON LA ENVOLVENTE.
C   LW = ESPACIO DE TRABAJO PARA REALIZAR EL FUNDIDO DE LA NUEVA LINEA
C   CON LA ENVOLVENTE.
C   ITIP = VARIABLE DE CONTROL DE ESCRITURA.
C   IER = VARIABLE DE CONTROL DE ERRORES.
C   IBDIS = VARIABLE DE CONTROL DE ESCRITURA.
C *****
C
C   LOGICAL LW
C   INTEGER ORDENA(400)
C   DIMENSION V(2,MAX),C(2,N),W(3,1),LW(1)
C
C   ORDENA = VECTOR ENTERO QUE SE UTILIZA EN LA CONSTRUCCION DE LA NUEVA
C   ENVOLVENTE.
C *****
C
C   DATA NPW/0/
C *****
C
C   ESCRITURA DE LA NUEVA LINEA A PINTAR.
C
C   IF (ITIP.NE.0)
C     *WRITE(6,401)
C 401 FORMAT(' C')
C     IF (ITIP.NE.0)
C     *WRITE(6,400) ((C(L1,L2),L1=1,2),L2=1,N)
C 400 FORMAT(' ' /5(6X,2F10.3))
C *****
C
C   IF(M.GT.0) GO TO 2
C *****
C
C   DIBUJO DE LA PRIMERA LINEA DEL DIAGRAMA.
C
C   MP=N
C   IF(MP.GT.MAX) MP=MAX
C   IP=3
C   DO 1 J=1,MP
C     V(1,J)=C(1,J)

```

```

V(2,1)=C(2,1)
CALL PLOTMM(C(1,1),C(2,1),IP)
IP=2
1 CONTINUE
C
C*****
RETURN
C*****
C
C*****
2 CONTINUE
C*****
C
C LLAMADA A LA SUBROUTINA DE FUNDIDO DE LA NUEVA LINEA CON LA ENVOLVEN-
C TE.
C
CALL MERGE(W,LW,NPM,V,M,C,N,KIN,KFN)
NPW=NPW+1
DO 11 J=2,NPM
C
C*****
IF(W(1,J).GE.W(1,J-1)) GO TO 11
C*****
C
C CACHE NO PUEDE PROCESAR LOS DATOS. ESTO PROVOCARA UNA INTERRUPCION
C EN EL PROGRAMA PRINCIPAL (POR EL VALOR QUE SE ALMACENA EN IER).
C
IF(NPW.LE.6)
*WRITE(6,100) ((W(I,J),I=1,3),LW(J),J=1,NPM)
WRITE(6,99)
99 FORMAT('0LOS DATOS ANTERIORES NO SON PROCESABLES!')
IER=NPW
C
C*****
RETURN
C*****
C
C*****
11 CONTINUE
C*****
C
IF(NPW.LE.IABS(IDBS))
*WRITE(6,100) ((W(I,J),I=1,3),LW(J),J=1,NPM)
100 FORMAT('11/3(6X,3F10.3,3X,L1))
IF(IDBS.GT.0)
*WRITE(6,101) KIN,KFN,NPM
101 FORMAT('0KIN, KFN, Y NPM = ',3I6)
C
C*****
C*****
C*****
C
C DIBUJO DE LAS SECCIONES VISIBLES.
C*****
IF(C(1,1).GT.V(1,1)) GO TO 12
CALL PLOTMM(V(1,1),V(2,1),3)
CALL PLOTMM(C(1,1),C(2,1),2)
12 CONTINUE

```

```

      K=0
      I=0
C
C      SIN P I N T A R
C 3 CONTINUE
      I=I+1
      IF(I.GT.NPM) GO TO 5
      IF(W(2,I).GT.W(3,I)) GO TO 3
      IF(I.GT.1) GO TO 13
      IF(IDBS.GT.0)
      *WRITE(6,200) I,W(1,I),W(3,I)
      CALL PLOTMM(W(1,I),W(3,I),3)
      GO TO 4
      13 CONTINUE
      CALL INTER(X,Y,W(1,I-1),W(1,I),1)
      IF(IDBS.GT.0)
      *WRITE(6,200) I,X,Y,W(1,I),W(3,I)
      200 FORMAT('OEMPEZANDO A PINTAR EN I = ',I3,2(6X,2F10.3))
      CALL PLOTMM(X,Y,3)
      CALL PLOTMM(W(1,I),W(3,I),2)
C EL PUNTO DE INTERSECCION PASARA A V.
      K=K+1
      C(1,K)=X
      C(2,K)=Y
C EL PUNTO DE INTERSECCION K ES ANTERIOR AL PUNTO I DE W.
      ORDENAK)=I
C
C      P I N T A N D O
C 4 CONTINUE
      I=I+1
      IF(I.GT.NPM) GO TO 5
      IF(W(2,I).GT.W(3,I)) GO TO 14
      CALL PLOTMM(W(1,I),W(3,I),2)
      GO TO 4
      14 CONTINUE
      CALL INTER(X,Y,W(1,I-1),W(1,I),2)
      IF(IDBS.GT.0)
      *WRITE(6,300) I,X,Y
      300 FORMAT('OACABANDO DE PINTAR EN I = ',I3,6X,2F10.3)
      CALL PLOTMM(X,Y,2)
C EL PUNTO DE INTERSECCION PASARA A V.
      K=K+1
      C(1,K)=X
      C(2,K)=Y
C EL PUNTO DE INTERSECCION K ES ANTERIOR AL PUNTO I DE W.
      ORDENAK)=I
      GO TO 3
      5 CONTINUE
      IF(V(1,M).GT.C(1,N)) GO TO 6
      CALL PLOTMM(C(1,N),C(2,N),3)
      CALL PLOTMM(V(1,M),V(2,M),2)
      6 CONTINUE
C
C*****
C*****
C*****
C
C ESCRITURA DE TODOS LOS PUNTOS DE INTERSECCION.

```

```

C
  IF (ITIP.NE.0)
    *WRITE(6,401)
    IF (ITIP.NE.0)
      *WRITE(6,400)((C(L1,L2),L1=1,2),L2=1,K)
C
C*****
C*****
C*****
C*****
C
C      CALCULO DE LA NUEVA LINEA DE VISIBILIDAD.
C*****
C      DO 61 L=1,K
        IF(C(1,L).GE.W(1,KIN)) GO TO 62
61 CONTINUE
62 CONTINUE
        K=K+1
        C(1,K)=999.
        C(2,K)=0
        MP=1
        V(1,MP)=W(1,KIN)
        V(2,MP)=AMAX1(W(2,KIN),W(3,KIN))
        I1=KIN+1
        I2=KFN-1
        DO 30 I=I1,I2
          IF(LW(I)) GO TO 10
          IF(W(3,I).LT.W(2,I)) GO TO 30
          Y=W(3,I)
          GO TO 20
10 CONTINUE
          IF(W(2,I).LT.W(3,I)) GO TO 30
          Y=W(2,I)
20 CONTINUE
21 CONTINUE
          IF(MP.GE.MAX) GO TO 30
          IF(C(1,L).GT.W(1,I)) GO TO 22
C SE PREGUNTA SI EL PUNTO DE INTERSECCION ES ANTERIOR O NO.
          IF(ORDENA(L).GT.I) GOTO 22
          MP=MP+1
          V(1,MP)=C(1,L)
          V(2,MP)=C(2,L)
          L=L+1
          GO TO 21
22 CONTINUE
          MP=MP+1
          V(1,MP)=W(1,I)
          V(2,MP)=Y
30 CONTINUE
          I=KFN
          IF(MP.GE.MAX) RETURN
C SE PREGUNTA SI YA HAN SIDO CONSIDERADOS TODOS LOS PUNTOS DE INTER-
C SECCION.
          IF((L.LT.K).AND.(C(1,L).LE.W(1,I))) GOTO 91
92 MP=MP+1
          V(1,MP)=W(1,I)
          V(2,MP)=AMAX1(W(2,I),W(3,I))
          GOTO 90

```



```

C
C *****
C *   ESTA SUBROUTINA INVESTIGA SI HAY SEGMENTOS PERTENECIENTES A PO- *
C *   *LIGONOS CONCENTRICOS. *
C *****
C
SUBROUTINE CHECK2(A,MAX,M,N)
DIMENSION A(MAX,N)
IF(M.EQ.1)RETURN
2 CONTINUE
I=0
1 CONTINUE
I=I+1
INCM=0
IF(A(I,2).LT.A(I+1,2)) GOTO 5
INDIC=0
DO 3 J=I,M-1
IF(A(J,2).LT.A(J+1,2)) GOTO 4
INDIC=INDIC+1
3 CONTINUE
4 CONTINUE
CALL CONCE2(A,MAX,M,N,I,INDIC,INCM)
M=M+INCM
CALL ORDEN(A,MAX,M,N,2)
GOTO 2
5 IF(I.EQ.M-1) RETURN
GOTO 1
C
C *****
C
END

```



```

C
C *****
C * SUBROUTINA DE CLASIFICACION POR CENTILES. EL USUARIO HA DEBIDO *
C *DEFINIR EL NUMERO DE CLASES QUE SE PRETENDEN. *
C *****
C
C SUBROUTINE CLASCT(N)
C
C N = NUMERO DE CASOS
C *****
C
C COMMON/A/P(100)
C COMMON/B/ITRAM(100)
C COMMON/D/ICLAS,IT(10)
C COMMON/E/UMB(9)
C DIMENSION IA(100)
C
C IA = VECTOR AUXILIAR PARA CONOCER EL ORDEN DE LOS CASOS A LA ENTRADA
C A ESTA SUBROUTINA.
C *****
C
C INICIALIZACION DE VALORES.
C
C DO 8 I=1,100
C 8 IA(I)=I
C *****
C
C SE ORDENAN LOS PESOS DE MENOR A MAYOR Y SE CONSERVAN EN IA SUS POSI-
C CIONES A LA ENTRADA A ESTA SUBROUTINA.
C
C N1=N-1
C DO 1 I=1,N1
C IMIN=I
C I1=I+1
C DO 2 II=I1,N
C IF(P(IMIN).GT.P(II))IMIN=II
C 2 CONTINUE
C V=P(IMIN)
C P(IMIN)=P(I)
C P(I)=V
C IORD=IA(IMIN)
C IA(IMIN)=IA(I)
C IA(I)=IORD
C 1 CONTINUE
C *****
C
C CLASIFICACION.
C
C ATENCION.SOLAMENTE CADA CENTIL REUNE UN MISMO NUMERO DE CASOS. CUAN-
C DO EL NUMERO DE CASOS DEL PROBLEMA ES MULTIPLO DEL NUMERO DE INTERVA
C LOS, EN CASO CONTRARIO EL CENTIL SUPERIOR RECDEGE EL EXCESO.
C
C NCC=N/ICLAS
C K=0

```

```

DO 7 I=1,ICLAS
J=0
3 CONTINUE
J=J+1
KJ=K+J
IIA=IA(KJ)
ITRAM(IIA)=IT(I)
IF(J.LT.NCC) GOTO 3
IF(I.EQ.ICLAS) GOTO 4
UMB(I)=(P(KJ)+P(KJ+1))/2.
GOTO 5
4 CONTINUE
IF(KJ.EQ.N) GOTO 5
KJ1=KJ+1
DO 6 IJK=KJ1,N
IIA2=IA(IJK)
ITRAM(IIA2)=IT(I)
6 CONTINUE
5 CONTINUE
K=KJ
7 CONTINUE
C
C*****
C
C Llamada a la subrutina que prepara la informacion necesaria para
C reproducir adecuadamente el diagrama de frecuencias y la leyenda.
C
C CALL TRAMLY(N,1)
C RETURN
C
C*****
C
END

```

```

C
C *****
C * SUBROUTINA DE CLASIFICACION POR UNIDADES DE DESVIACION STANDARD.*
C *****
C
C SUBROUTINE CLASDT(N,RIDT)
C
C N = NUMERO DE CASOS
C RIDT = 1. LA AMPLITUD DE LAS CLASES ES DE 1 DESVIACION STANDARD.
C RIDT = 2. LA AMPLITUD DE LAS CLASES ES DE 1/2 DESVIACION STANDARD.
C *****
C
C COMMON/A/P(100)
C COMMON/B/ITRAM(100)
C COMMON/D/ICLAS,IT(10)
C COMMON/E/UMB(9)
C *****
C
C SE CALCULA LA DESVIACION TIPICA DE LA VARIABLE SOBRE LOS CASOS DEL
C PROBLEMA.
C
C RMEDIA=AVER(N)
C DESTP =DESTIP(N)
C *****
C
C SE TRANSFORMAN TODOS LOS PESOS PARA EXPRESARLOS EN UNIDADES DE DES-
C VIACION STANDARD.
C
C DO 1 I=1,N
C P(I)=(P(I)-RMEDIA)/DESTP
C 1 CONTINUE
C *****
C
C SE CALCULAN LOS UMBRALES DE CLASIFICACION.
C
C IBASE= -ICLAS/2
C BASE=FLOAT(IBASE)/RIDT
C UMB(1)=BASE+1./RIDT
C DO 2 I=2,ICLAS-1
C 2 UMB(I)=UMB(I-1)+1./RIDT
C *****
C
C CLASIFICACION.
C
C DO 4 I=1,N
C DO 5 J=1,ICLAS-1
C IF(P(I).LE.UMB(J)) GOTO 6
C 5 CONTINUE
C ITRAM(I)=IT(ICLAS)
C GOTO 4
C 6 CONTINUE
C ITRAM(I)=IT(J)
C 4 CONTINUE

```

```
C
C*****
C SE TRANSFORMAN LOS UMBRALES A LAS UNIDADES DE PARTIDA, ANTES DE LLA-
C MAR A TRAMLY.
C
C   DO 7 I=1,ICLAS-1
C   7 UMB(I)=UMB(I)*DESTP+RMEDIA
C*****
C LLAHADA A LA SUBROUTINA QUE PREPARA LA INFORMACION NECESARIA PARA
C REPRODUCIR ADECUADAMENTE EL DIAGRAMA DE FRECUENCIAS Y LA LEYENDA.
C
C   CALL TRAMLY(N,1)
C   RETURN
C*****
C
C   END
```

```

C
C *****
C * SUBROUTINA DE CLASIFICACION STANDARD DE LOS CASOS EN SEIS INTER-
C * VALOS DE IGUAL AMPLITUD.
C *****
C
C SUBROUTINE CLASST(N)
C
C N = NUMERO DE CASOS
C *****
C
C COMMON/A/P(100)
C COMMON/B/ITRAM(100)
C COMMON/D/ICLAS,IT(10)
C COMMON/E/UMB(9)
C
C ITRAM = VECTOR ENTERO DONDE SE ALMACENARA MEDIANTE UN CODIGO EL
C TIPO DE TRAMA QUE CORRESPONDE A CADA AREA.
C *****
C
C TRAMAS STANDARD.
C
C IT(1)=1
C IT(2)=2
C IT(3)=5
C IT(4)=6
C IT(5)=16
C IT(6)=22
C *****
C
C SE INVESTIGA EL RANGO DE LA VARIABLE Y SE DEFINEN LOS CINCO UMBRALES
C
C PMAX=-1.E74
C PMIN= 1.E74
C DO 1 I=1,N
C IF(PMAX.LE.P(I)) PMAX=P(I)
C IF(PMIN.GE.P(I)) PMIN=P(I)
C 1 CONTINUE
C RANG=PMAX-PMIN
C AM=RANG/6.
C DO 2 I=1,5
C UMB(I)=PMIN+AM*I
C 2 CONTINUE
C *****
C
C SE CLASIFICAN LOS CASOS.
C
C DO 3 I=1,N
C DO 4 J=1,5
C IF(P(I).LE.UMB(J)) GOTD5
C 4 CONTINUE
C ITRAM(I)=IT(6)
C GOTD 3
C 5 CONTINUE

```

```
      ITRAM(I)=IT(J)
3  CONTINUE
      ICLAS=6
C *****
C
C LLAMADA A LA SUBROUTINA QUE PREPARA LA INFORMACION NECESARIA PARA
C REPRODUCIR ADECUADAMENTE EL DIAGRAMA DE FRECUENCIAS Y LA LEYENDA.
C
      CALL TRAMLY(N,I)
      RETURN
C *****
C
      END
```

```

C
C *****
C * SUBROUTINA DE CLASIFICACION POR UMBRALES QUE HA DEFINIDO EL USUA*
C *RIO. EL NUMERO DE CLASES NO PUEDE SER NUNCA SUPERIOR A 10. *
C *****
C
C SUBROUTINE CLASUM(N)
C
C N = NUMERO DE CASOS
C *****
C
C COMMON/A/P(100)
C COMMON/B/ITRAM(100)
C COMMON/D/ICLAS,IT(10)
C COMMON/E/UMB(9)
C *****
C
C DO 1 I=1,N
C DO 2 J=1,ICLAS-1
C IF(P(I).LE.UMB(J)) GOTO 3
C 2 CONTINUE
C ITRAM(I)=IT(ICLAS)
C GOTO 1
C 3 CONTINUE
C ITRAM(I)=IT(J)
C 1 CONTINUE
C *****
C
C LLAMADA A LA SUBROUTINA QUE PREPARA LA INFORMACION NECESARIA PARA
C REPRODUCIR ADECUADAMENTE EL DIAGRAMA DE FRECUENCIAS Y LA LEYENDA.
C
C CALL TRAMLY(N,1)
C RETURN
C *****
C
C END

```

```

C
C *****
C * ESTA SUBROUTINA CAMBIA EL ORDEN DE LOS PUNTOS QUE DESCRIBEN LA *
C * ENVOLVENTE. *
C *****
C
C SUBROUTINE CLUCK(N,NIE)
C
C COMMON/F/P(100,100),E(1000,2),A(50,2),DISMAX
C COMMON/G/LP(50),H(50),NP,LE,INFUR,OUT
C COMMON/H/IE(1000),IA(50)
C
C INTEGER OUT,P,E,A,DISMAX,AUX1,AUX2
C
C DO 1 I=1,N
C   AUX1=E(LE,1)
C   AUX2=E(LE,2)
C
C DO 2 J=1,LE-1
C   IJ=LE-(J-1)
C   IJM1=IJ-1
C   E(IJ,1)=E(IJM1,1)
C   E(IJ,2)=E(IJM1,2)
C 2 CONTINUE
C   E(1,1)=AUX1
C   E(1,2)=AUX2
C
C DO 3 K=2,NIE
C   IK=NIE-(K-2)
C   IKM1=IK-1
C   IE(K)=IE(IKM1)+1
C 3 CONTINUE
C
C 1 CONTINUE
C   RETURN
C *****
C
C END

```



```

C
C *****
C *   ESTA SUBROUTINA EFECTUA UN CLUSTER DE 'MAXIMA SIMILITUD' DE 'NC'*
C *CASOS, EN FUNCION DE 'NV' VARIABLES, QUE DA COMO RESULTADO UN AGRU*
C *PAMIENTO EN 'ICLAS' CLASES.
C *****
C
C   SUBROUTINE CLUSTR(NC,NV,ISW)
C *****
C
C   COMMON/B/ITRAM(100)
C   COMMON/C/P(100,10)
C   COMMON/D/ICLAS,IT(10)
C   COMMON/G/D(100,100)
C
C   D = MATRIZ DE DISTANCIAS ENTRE LOS CASOS DEL PROBLEMA.
C *****
C
C   IF(ISW.EQ.5) GOTO 10
C *****
C
C   ISW=4. EN INPUT SE HAN LEIDO LOS PESOS DE UN MAXIMO DE 10 VARIABLES
C   SOBRE UN MAXIMO DE 100 CASOS, EN LA MATRIZ P(100,10). AHORA SE VAN
C   A CALCULAR LAS DISTANCIAS EUCLIDEAS ENTRE CASOS EN UN ESPACIO DE
C   TANTAS DIMENSIONES COMO VARIABLES. ESTOS VALDRES SE ALMACENARAN EN
C   LA MATRIZ D(100,100).
C
C   DO 1 I=1,NC
C     DO 2 J=1,NC
C       SD2=0.
C       DO 3 K=1,NV
C         D2=(P(I,K)-P(J,K))*(P(I,K)-P(J,K))
C         SD2=SD2+D2
C       3 CONTINUE
C       RSD2=SQRT(SD2)
C       D(I,J)=RSD2
C     2 CONTINUE
C   1 CONTINUE
C *****
C
C   10 CONTINUE
C
C   ISW=5. EN INPUT (A TRAVES DE TRANS2) SE HA LEIDO O CALCULADO UNA
C   MATRIZ DE DISTANCIAS.
C *****
C
C   INICIALIZACION DE VALDRES.
C
C   INC=NC
C   DO 11 I=1,NC
C     INC=INC+1
C     ITRAM(I)=INC
C   11 CONTINUE
C

```

```

C      NT=1
C*****
C*****
C*****
C
C  COMIENZA LA ITERACION QUE VA A DAR COMO RESULTADO LA DEFINICION DE
C  UN NUMERO DE GRUPOS IGUAL A ICLAS. EL PROCEDIMIENTO UTILIZADO ES EL
C  DE 'MAXIMA SIMILITUD'.
C
C 12 CONTINUE
C 12.1. SE BUSCA LA DISTANCIA MINIMA EN LA MATRIZ D.
C
C      DMIN=1.E74
C      DO 13 I=2,NC
C      DO 14 J=1,I-1
C      IF(D(I,J).LE.DMIN) GOTO15
C      GOTO 14
C 15 CONTINUE
C      DMIN=D(I,J)
C      IMEM=I
C      JMEM=J
C 14 CONTINUE
C 13 CONTINUE
C      D(IMEM,JMEM)=1.E75
C
C 12.2. SE AGRUPAN LOS DOS CASOS MAS PROXIMOS ENTRE SI.
C
C      IF(ITRAM(IMEM).LE.NC.AND.ITRAM(JMEM).LE.NC) GOTO 16
C      IF(ITRAM(IMEM).LE.NC) GOTO 17
C      IF(ITRAM(JMEM).LE.NC) GOTO 18
C
C 12.2.1. NO HA SIDO PREVIAMENTE CONSIDERADO NINGUNO DE LOS DOS CASOS.
C
C      ITRAM(IMEM)=NT
C      ITRAM(JMEM)=NT
C      GOTO 19
C
C 12.2.2. HA SIDO YA CONSIDERADO EN OTRA AGRUPACION ANTERIOR EL CASO I.
C
C 17 CONTINUE
C      ITRAM(JMEM)=ITRAM(IMEM)
C      NT=NT-1
C      GOTO 19
C
C 12.2.3. HA SIDO YA CONSIDERADO EN OTRA AGRUPACION ANTERIOR EL CASO J.
C
C 18 CONTINUE
C      ITRAM(IMEM)=ITRAM(JMEM)
C      NT=NT-1
C      GOTO 19
C
C 12.2.4. AMBOS CASOS HAN SIDO YA CONSIDERADOS EN ANTERIORES AGRUPA-
C  MIENTOS.
C
C 16 CONTINUE
C

```

```

      IF(ITRAM(IMEM).EQ.ITRAM(JMEM)) GOTO 25
C
C 12.2.4.1.LOS DOS CASOS FORMABAN PARTE DE GRUPOS DISTINTOS.
C
      ITRAM1=ITRAM(IMEM)
      ITRAM2=ITRAM(JMEM)
      IREF1 =MIN(ITRAM1,ITRAM2)
      IREF2 =MAX(ITRAM1,ITRAM2)
      DO 20 I=1,NC
      IF(ITRAM(I).EQ.ITRAM1.OR.ITRAM(I).EQ.ITRAM2) ITRAM(I)=IREF1
20 CONTINUE
      DO 21 I=1,NC
      IF(ITRAM(I).LE.NC.AND.ITRAM(I).GT.IREF2) ITRAM(I)=ITRAM(I)-1
21 CONTINUE
      NREF=0
      DO 22 I=1,NC
      IF(ITRAM(I).GT.NC) GOTO 22
      NREF=MAX(NREF,ITRAM(I))
22 CONTINUE
      NT=NREF
      GOTO 19
C
C 12.2.4.2.LOS DOS CASOS FORMABAN PARTE DEL MISMO GRUPO.
C
      25 CONTINUE
      NT=NT-1
C
C*****
C 19 CONTINUE
C*****
      NT=NT+1
C
C 12.3.TEST PARA DECIDIR SI ES NECESARIO SEGUIR AGRUPANDO CASOS.
C
      ICONT=0
      DO 40 I=1,NC
      IF(ITRAM(I).LE.NC) GOTO 40
      ICONT=ICONT+1
40 CONTINUE
      IF((ICONT+NT-1).GT.ICLAS) GOTO 12
C
      ITA=NT
      DO 41 I=1,NC
      IF(ITRAM(I).LE.NC) GOTO 41
      ITRAM(I)=ITA
      ITA=ITA+1
41 CONTINUE
C
C*****
C*****
C SE SITUA AHORA EN ITRAM LA INFORMACION NECESARIA PARA QUE TENGAN LA
C MISMA TRAMA LOS CASOS DEL MISMO GRUPO, Y PARA QUE LA GAMA DE TRAMAS
C SEA LA ELEGIDA POR EL USUARIO.

```

```
C
  DO 24 I=1,NC
    ITRAM(I)=IT(ITRAM(I))
  24 CONTINUE
C
C.....
C
  CALL TRAMLY(NC,2)
C
C  YA ESTA PREPARADA LA ZONA COMUN F PARA LLAMAR AL HISTOGRAMA (NO A
C  LA LEYENDA, QUE EN ESTE CASO NO DEBE UTILIZARSE)
C
  RETURN
C
C.....
C
  END
```

```

C
C *****
C * SUBROUTINE DE CONTROL DE LA PLUMA EN LOS DIVERSOS DIBUJOS.*
C *****
C
C SUBROUTINE CNPEN(X,Y)
C
C (X,Y) = VERTICE SUPERIOR DERECHO DEL RECTANGULO QUE ENMARCA UN
C DIBUJO.
C *****
C
C COMMON/TRID,XV(300),YV(300),B(2,100),XOR,YOR,FCT,VX,VY,VZ,WR,DR
C ,DS,ZB,ZO,ZI,ZF,ZINC,N
C CHARACTER*56 MENSA
C CHARACTER RN,RESP
C *****
C
C INICIALIZACION DE VALORES.
C
C DATA XT,YT,YSHEET,XSHEET,XSEND/0.,0.,81.,130.,0./
C DATA RN,RESP/'R',' '/
C DATA MENSA/'MONTAR OTRA HOJA,POR FAVOR.PARA CONTINUAR,RESPONDER'
C ' '/
C
C XT CONTROLA LA MAYOR DIMENSION EN X DE TODOS LOS DIBUJOS DE UNA
C RISTRA VERTICAL. YT VA SUPERPONIENDO ADECUADAMENTE UNOS DIBUJOS
C SOBRE OTROS. YSHEET ES EL TAMAÑO DEL PLIEGO EN LA DIMENSION Y.
C XSHEET ES EL TAMAÑO DEL PLIEGO EN LA DIMENSION X. XSEND ES UN
C ACUMULADOR QUE SE UTILIZA PARA CONOCER CUANDO HAY QUE CAMBIAR
C DE PLIEGO.
C *****
C
C CALL FACTOR(1.)
C
C XT=AMAX1(XT,X)
C YT=YT+Y
C IF((YT+Y).GT.YSHEET) GOTO 1
C CALL PLOT(0.,(Y+2.),-3)
C YT=YT+2.
C RETURN
C
C 1 CONTINUE
C
C SE PRESUPONE QUE NO VA A CABER EL SIGUIENTE DIBUJO Y SE EMPIEZA
C UNA NUEVA RISTRA VERTICAL.
C
C XSEND=XSEND+(XT+2)
C IF((XSEND+XT).GT.XSHEET) GOTO 2
C CALL PLOT((XT+2.),-(YT-Y),-3)
C XT=0.
C YT=0.
C RETURN
C
C 2 CONTINUE

```

```
C SE PRESUPONE QUE NO VA A CABER EL SIGUIENTE DIBUJO EN EL PLIEGO  
C Y SE PIDE UN NUEVO PLIEGO.  
C
```

```
DO 3 NIN=1,100  
CALL PLOT(0.,0.,3)  
CALL PLOT(0.,0.5,3)  
3 CALL PLOT(0.,0.,3)  
CALL SEND(RN,MENSA,RESP)  
XT=0.  
YT=0.  
XSEND=0.  
RETURN
```

```
C  
C .....  
C  
END
```

```

C
C *****
C *   ESTA SUBROUTINA RESUELVE LOS PROBLEMAS QUE PLANTEA PARA EL DIBU-
C *   *JO LA EXISTENCIA DE POLIGONOS CONCENTRICOS.
C *****
C
SUBROUTINE CONCE2(A,MAX,M,N,IPUNT,INDIC,INCM)
DIMENSION A(MAX,N),B(50,3)
DATA INFOR/1/
LB=1+2*INDIC
IM=0
DO 1 I=1,LB
IM1=I-1
IF(I.GT.INDIC) GOTO 2
B(I,1)=A(IPUNT+IM1,1)
B(I,2)=A(IPUNT+I,1)
B(I,3)=A(IPUNT+IM1,3)
GOTO 1
2 CONTINUE
IF(I.GT.INDIC+1) GOTO 3
B(I,1)=A(IPUNT+INDIC,1)
B(I,2)=A(IPUNT+INDIC,2)
B(I,3)=A(IPUNT+INDIC,3)
GOTO 1
3 CONTINUE
B(I,1)=A(IPUNT+INDIC-IM,2)
B(I,2)=A(IPUNT+INDIC-(IM+1),2)
B(I,3)=A(IPUNT+INDIC-(IM+1),3)
IM=IM+1
1 CONTINUE
IF(INFOR.NE.1)
*WRITE(6,102) ((B(I,J),J=1,3),I=1,LB)
102 FORMAT(/'0B'/(5X,3F5.1))
C
C SE DEJA ESPACIO PARA SITUAR ADECUADAMENTE, EN LA LISTA, LOS NUEVOS
C SEGMENTOS.
C
IDESPL= LB-(INDIC+1)
DO 4 I=1,M
IM1=I-1
IF((M-IM1).EQ.(IPUNT+INDIC)) GOTO 5
DO 6 J=1,3
A(M-IM1+IDESPL,J)=A(M-IM1,J)
6 CONTINUE
4 CONTINUE
C
C SE TRASLADAN LOS NUEVOS SEGMENTOS DE B A A.
C
5 CONTINUE
IF(INFOR.NE.1)
*WRITE(6,103)((A(I,J),J=1,3),I=1,M+IDESPL)
103 FORMAT(/'0A'/(5X,3F5.1))
DO 7 I=1,LB
IM1=I-1
DO 8 J=1,3
A(IPUNT+IM1,J)=B(I,J)
8 CONTINUE
7 CONTINUE

```

301

```
IF(INFOR.NE.1)
*WRITE(6,103)((A(I,J),J=1,3),I=1,M+IDESPL)
IPUNT=IPUNT+(LB-1)
INCM=IDESPL
RETURN
```

```
C
C*****
C
END
```



```

C
C *****
C * ESTA SUBROUTINA AVERIGUA QUE POLIGONO, DEFINIDO POR LAS COORDE- *
C * NADAS DE SUS VERTICES, TIENE UNA FRONTERA COMUN CON OTRO DEFINIDO *
C * ANTERIORMENTE. *
C *****
C
C SUBROUTINE CONTIG(I, I1, I2, I3, I4, NUMAS)
C
C I = NUMERO DE ORDEN DEL POLIGONO.
C I1 E I2 = NUMEROS DE ORDEN EN LA LISTA DE SUS VERTICES DE LOS VERTI-
C CES QUE INICIAN Y CIERRAN LA SECCION COMUN CON EL OTRO POLIGONO.
C I3 E I4 = NUMEROS DE ORDEN EN LA LISTA DE LOS VERTICES DEL OTRO PO-
C LIGONO DE LOS VERTICES QUE INICIAN Y CIERRAN LA SECCION COMUN.
C NUMAS = VARIABLE LOGICA, ESTA SUBROUTINA DEVUELVE EN ELLA EL VALOR
C .TRUE. CUANDO NO ENCUENTRA NINGUN POLIGONO FRONTERIZO.
C *****
C
C COMMON/FP/(100,100),E(1000,2),A(50,2),DISMAX
C COMMON/G/LP(50),IC(50),NP,LE,INFOR,IUT
C COMMON/H/IE(1000),IA(50)
C COMMON/L/RESTO
C LOGICAL RESTO,OTRO,NUMAS,SICLOC,SIOTH
C INTEGER IUT,P,E,A,DISMAX,PRUX,PRIXI
C
C LAS POSICIONES DE MEMORIA DE LAS ZONAS COMUNES F Y G ESTAN DESCRITAS
C EN POLIGS4.
C IE = VECTOR ENTERO DONDE SE ALMACENAN LOS NUMEROS DE ORDEN DE LOS
C VERTICES DE LA FRONTERA DE LA ENVOLVENTE.
C IA = VECTOR ENTERO DONDE SE ALMACENAN LOS NUMEROS DE ORDEN DE LOS
C VERTICES DE LA FRONTERA DEL POLIGONO A FUNDIR.
C RESTO = VARIABLE LOGICA QUE TOMA EL VALOR .TRUE. CUANDO LA LISTA DE
C VERTICES DEL POLIGONO EN LA FRONTERA 'PASA POR 0'.
C OTRO = VARIABLE LOGICA QUE TOMA EL VALOR .TRUE. CUANDO EL TIPO DE
C FRONTERA QUE SE PRODUCE NO ES TRATABLE POR EL PROGRAMA. ESTO
C PROVOCA QUE EL PROGRAMA BUSQUE UN NUEVO POLIGONO A FUNDIR CON
C LA ENVOLVENTE.
C NUMAS = VARIABLE LOGICA QUE TOMA EL VALOR .TRUE. CUANDO TODOS LOS
C POLIGONOS YA HAN SIDO FUNDIDOS.
C SICLOC = VARIABLE LOGICA QUE TOMA EL VALOR .TRUE. CUANDO LA FRON-
C TERA 'PASA POR 0' EN LA ENVOLVENTE.
C SIOTH = VARIABLE LOGICA QUE TOMA EL VALOR .TRUE. CUANDO EL NUEVO
C POLIGONO A FUNDIR PUSEE TODOS LOS PUNTOS DE LA ENVOLVENTE
C HASTA ESE MOMENTO.
C *****
C
C NCONT=0
C *****
C 6 CONTINUE
C *****
C NCONT=NCONT+1
C
C DO 1 I=1,NP
C IF(N(I).NE.0) GOTO 1
C I2=I+2
C I2M1=I2-1

```

```

C*****
8 CONTINUE
C*****
NODOS=0
DO 2 J=1,LE
DO 3 K=1,LP(I)
PRUX=IABS(E(J,1)-P(K,I2M1))
PRUX1=IABS(E(J,2)-P(K,I2))
IF(PRUX.LT.DISHAX)
*AND,PRUX1.LT.DISHAX) GOTO 4
GOTO 3
4 CONTINUE
NODOS=NODOS+1
IE(NODOS)=J
IA(NODOS)=K
3 CONTINUE
2 CONTINUE
IF((NODOS.EQ.0).OR.(NODOS.EQ.1)) GOTO 1
C
C EL POLIGONO I TIENE FRONTERA CON LA ENVOLVENTE.
C
IF(INFOR.EQ.1)
*WRITE(OUT,100)(IE(I43),I43=1,NODOS)
C
C SE GARANTIZA QUE LA FRONTERA COMUN EN LA ENVOLVENTE 'NO PASE POR 0'
C Y QUE LA FRONTERA SEA 'CONTIGUA'.
C
SICLOC=.FALSE.
OTRO=.FALSE.
CALL TESTFE(NODOS,OTRO,SICLOC)
C
*****
IF(OTRO) GOTO 1
IF(INFOR.EQ.1)
*WRITE(OUT,100)(IE(I43),I43=1,NODOS)
C
I3=IE(1)
I4=IE(NODOS)
C
IF(INFOR.EQ.1)
*WRITE(OUT,100)(IA(I43),I43=1,NODOS)
C
C SI LA FRONTERA 'PASA POR 0' EN LA ENVOLVENTE, ES NECESARIO MODIFI-
C CAR EL ORDEN DE DESCRIPCION DE LOS PUNTOS COMUNES DEL POLIGONO A
C FUNDIR.
C
IF(SICLOC) CALL NEWAI
*****
IF(INFOR.EQ.1)
*WRITE(OUT,100)(IA(I43),I43=1,NODOS)
C
C ESTE TRATAMIENTO SOLUCIONA EL PROBLEMA DE FORMACION DE LA NUEVA EN-
C VOLVENTE, CUANDO EL NUEVO POLIGONO A FUNDIR POSEE TODOS LOS PUNTOS
C DE LA ENVOLVENTE ANTERIOR.
C
SIOTH=.FALSE.
IF(NODOS.EQ.LE) CALL OTHWAY(NODOS,SIOTH,I)
*****
IF(SIOTH) GOTO 8

```

```

      RESTO=.FALSE.
C
C  SI LA FRONTERA COMUN EN EL POLIGONO 'PASA POR O' RESTO =.TRUE.
C
      CALL TESTFAC(NODOS,I)
      *****
      I1=IA(I)
      I2=IA(FRONS)
C
C  SE VA A FUNDIR EL POLIGONO I.
C
      M(I)=1
C*****
      RETURN
C*****
      1 CONTINUE
C
      IF(I;CONT.GT.NP) GOTO 7
C
      DO 5 IT=1,NP
      IF(M(IT).EQ.0) GOTO 6
      5 CONTINUE
C
      NO SE HA ENCONTRADO NINGUN POLIGONO FRONTERIZO,
C
      NOMAS=.TRUE.
C*****
      RETURN
C*****
C
      NO SE PUEDEN FUNDIR TODOS LOS POLIGONOS.
C
      7 CONTINUE
      WRITE(OUT,101)
      CALL EXIT
      *****
C*****
C
      100 FIRMAT(1014)
      101 FORMAT(' NO PUEDE FUNDIR TODOS LOS POLIGONOS')
      END

```

```

C
C *****
C *  ESTA SUBROUTINA LEE LAS COORDENADAS DE N CONTORNOS, LOS DIBUJA *
C *Y LOS TRAMA COHERENTEMENTE CON EL PESO DE UNA VARIABLE SOBRE CADA *
C *UNO DE ELLOS. *
C *****
C
C SUBROUTINE COROPL(N,FCT)
C
C N = NUMERO DE AREAS A DIBUJAR Y TRAMAR.
C FCT = FACTOR DE ESCALA DEL DIBUJO.
C *****
C
C COMMON/B/ITRAM(100)
C DIMENSION XY(100,2)
C CHARACTER FMT(80)
C
C XY = MATRIZ DONDE VAN A ALMACENARSE SUCESIVAMENTE LAS COORDENADAS
C QUE DESCRIBEN CADA CONTORNO. SI LOS CONTORNOS ESTAN DESCRITOS
C POR MAS DE 100 PUNTOS, HAY QUE REDIMENSIONAR ESTA MATRIZ.
C FMT = VECTOR DE CARACTERES PARA DESCRIBIR EL FORMATO DE LAS COORDE-
C NADAS.
C *****
C
C SE LEE EL FORMATO DE LAS COORDENADAS.
C
C READ(5,101) (FMT(I),I=1,80)
101 FORMAT(80A1)
C
C MAX=100
C
C DO 1 I=1,N
C
C SE LEE EL NUMERO DE PUNTOS QUE DESCRIBEN CADA CONTORNO.
C
C READ(8,100) L
100 FORMAT(T7,I2)
C IP=3
C LM1=L+1
C DO 2 J=1,LM1
C IF(J.E0,LM1) GOTO 3
C
C SE LEEN LAS COORDENAS DE LOS PUNTOS DE CADA CONTORNO.
C
C READ(9,FMT) XY(J,1),XY(J,2)
C XY(J,1)=XY(J,1)*FCT
C XY(J,2)=XY(J,2)*FCT
3 CONTINUE
C X=XY(J,1)/FCT
C Y=XY(J,2)/FCT
C IF(J.E0,LM1)X=XY(1,1)/FCT
C IF(J.E0,LM1)Y=XY(1,2)/FCT
C
C SE DIBUJA CADA CONTORNO.
C
C CALL PLOTM2(X,Y,IP)

```

```
      1P=2
      2 CONTINUE
      CALL WHERE(XPAGE,YPAGE,RFACT)
      CALL FACTOR(1.)
C
C
C      SE TRAMA CADA CONTORNO.
      CALL TRAMA(XY,MAX,L,ITRAM(1))
      CALL FACTOR(RFACT)
      1 CONTINUE
C
C*****
C
C      RETURN
C*****
C
C      END
```

```

C
C *****
C *   ESTA SUBROUTINA DIBUJA UN CUADRADO DE LADO D CUYO VERTICE INFE- *
C *RIOR IZQUIERDO SE SITUA EN (X,Y), ALMACENANDO LAS COORDENADAS DE *
C *LOS CUATRO VERTICES EN LA ZONA COMUN E. *
C *****
C
SUBROUTINE CUADRA(X,Y,D)
COMMON/E/XS(4,180),YS(4,180)
CALL PLOT (X,Y,3)
DO 1 I=1,4
GOTO(2,3,4,5),I
2 X=X+D
GOTO 6
3 Y=Y+D
GOTO 6
4 X=X-D
GOTO 6
5 Y=Y-D
6 CONTINUE
XS(I,1)=X
YS(I,1)=Y
CALL PLOT(X,Y,2)
1 CONTINUE
C
C *****
C
RETURN
END

```

```

C
C *****
C *   ESTA SUBROUTINA DIBUJA UN CUADRADO DE LADO D CON VERTICE INFE- *
C *RIOR IZQUIERDO EN (X,Y), ALMACENANDO EN LA MATRIZ XYS LAS COORDE- *
C *NADAS DE LOS CUATRO VERTICES. *
C *****
C
  SUBROUTINE CUADRA(X,Y,XYS,D)
    DIMENSION XYS(4,2)
    CALL PLOT(X,Y,3)
    DO 1 I=1,4
      GOTO(2,3,4,5),I
    2 X=X+D
      GOTO 6
    3 Y=Y+D
      GOTO 6
    4 X=X-D
      GOTO 6
    5 Y=Y-D
    6 CONTINUE
      XYS(I,1)=X
      XYS(I,2)=Y
      CALL PLOT(X,Y,2)
    1 CONTINUE
      RETURN
C
C *****
C
  END

```

```

C
C *****
C *   ESTA SUBROUTINA SIMPLIFICA LA DESCRIPCION POR PUNTOS DE UNA LI*
C *NEA, IGNORANDO UNO DE CADA DOS PUNTOS QUE SE ENCUENTREN SUFICIENTE*
C *TEMENTE PROXIMOS -CUANDO SUS COORDENADAS DIFIEREN, TAN SOLO, EN LA*
C *DIEZMILESIMA DE CM-. *
C *****
C
SUBROUTINE DEPURA(L)
COMMON/A/COORD(200,100),POLY(500,2)
J=0
DO 1 I=1,L
IM1=I+1
IF(1.E0.L)IM1=1
IF(ABS(POLY(I,1)-POLY(IM1,1)).LT.0.0001.
*AND,ABS(POLY(I,2)-POLY(IM1,2)).LT.0.0001)GOTO 1
J=J+1
POLY(J,1)=POLY(I,1)
POLY(J,2)=POLY(I,2)
1 CONTINUE
L=J
RETURN
C
C *****
C
END

```



```

C
C *****
C * ESTA SUBROUTINA DIBUJA EL DIAGRAMA DE FRECUENCIAS DE LA CLASIFI-
C * CACION PREVIA -SEA CUAL SEA EL PROCEDIMIENTO UTILIZADO-, SOBRE UN *
C * RECTANGULO DE DIMENSIONES XL*YL, CUYO ANGULO INFERIOR IZQUIERDO SE
C * SITUA EN (XO,YO).
C * ESTA SUBROUTINA NO ALTERA EL ORIGEN DE COORDENADAS DEFINIDO
C * ANTES DE LLAMARLA.
C *****
C
C SUBROUTINE DIAGRM(XO,YO,XL,YL)
C *****
C
C COMMON/D/ICLAS,IT(10)
C COMMON/F/IFR(10,2),UMBIS(10,2)
C DIMENSION XYS(4,2)
C
C XYS= MATRIZ QUE SE UTILIZA EN EL TRAMADO DE LAS BARRAS DEL DIA-
C GRAMA.
C *****
C
C TRASLADO DEL ORIGEN DE COORDENADAS AL ANGULO INFERIOR IZQUIERDO
C DEL RECTANGULO AL QUE SE VA A AJUSTAR EL DIBUJO DEL DIAGRAMA.
C
C CALL PLOT(XO,YO,-3)
C *****
C
C *CALCULO DE LOS PARAMETROS DE DIBUJO*
C *****
C
C 1. BASE DE LAS BARRAS DEL DIAGRAMA ( = MARGEN LATERAL).
C
C B=XL/FLOAT(ICLAS+2)
C
C 2. MARGENES SUPERIOR E INFERIOR.
C
C YM=YL/10.
C
C 3. ESCALA VERTICAL DEL DIAGRAMA = NUMERO DE CM DEL PLANO DE DIBUJO
C POR UNIDAD DE FRECUENCIA.
C
C MAX=0
C DO 1 I=1,ICLAS
C IF(MAX.LE.IFR(I,2))MAX=IFR(I,2)
C 1 CONTINUE
C ALT=YL-2*YM
C ESC=ALT/FLOAT(MAX)
C *****
C
C SE DIBUJAN LOS EJES DEL DIAGRAMA Y SE ROTULAN ADECUADAMENTE.
C
C CALL PLOT(B,(YM+ALT),3)
C CALL PLOT(B,YM,2)
C ESCI=1./ESC

```

```

TAM=AMIN1(0.3,B/6.)
I=0
ROTUL=0.
2 CONTINUE
I=I+1
IF(MOD(I,2).NE.0)
*CALL NUMBER((B-6*TAM),YM+FLOAT(I-1),TAM,ROTUL,0.,2)
CALL PLOT((B-TAM),YM+FLOAT(I-1),3)
CALL PLOT(B,YM+FLOAT(I-1),2)
ROTUL=ROTUL+ESC1
IF(FLOAT(I).GT.ALT) GOTO 3
GOTO 2
3 CONTINUE
CALL PLOT(B,YM,3)
CALL PLOT((XL-B),YM,2)
I=0
ROTUL=0.
4 CONTINUE
I=I+1
IF(MOD(I,2).NE.0)
*CALL NUMBER((B+(ICLAS+1)*TAM),YM+FLOAT(I-1),TAM,ROTUL,0.,2)
CALL PLOT((B+(ICLAS+1)*TAM),YM+FLOAT(I-1),3)
CALL PLOT((B+(ICLAS+1)),YM+FLOAT(I-1),2)
ROTUL=ROTUL+ESC1
IF(FLOAT(I).GT.ALT) GOTO 5
GOTO 4
5 CONTINUE
CALL PLOT((XL-B),(YM+ALT),3)
CALL PLOT((XL-B),YM,2)

```

```

C
C*****
C
C SE DIBUJAN LAS BARRAS Y SE TRAMAN COHERENTEMENTE.
C

```

```

X1=0.
DO 6 I=1,ICLAS
IP=3
X1=X1+B
Y=YM
Y1=IFR(I,2)*ESC
DO 7 J=1,4
GOTO(8,9,10,11),J
8 X=X1
GOTO 12
9 Y=Y+Y1
GOTO 12
10 X=X+B
GOTO 12
11 Y=Y-Y1
12 CALL PLOT(X,Y,IP)
XYS(J,1)=X
XYS(J,2)=Y
IP=2
7 CONTINUE
CALL WHERE(XPAGE,YPAGE,RFACT)
CALL FACTOR(1.)
CALL TRAMA(XYS,4,4,IFR(I,1))
CALL FACTOR(RFACT)

```

```
      6 CONTINUE
C
C*****
C
C  ESTA SENTENCIA DEVUELVE EL ORIGEN DE COORDENADAS A DONDE ESTABA
C  ANTES DE LLAMAR A DIAGRM.
C
C      CALL PLOT(-X0,-Y0,-3)
C      RETURN
C*****
C
C      END
```

```

C
C *****
C * SUBROUTINA DE DIBUJO DE PIRAMIDES. *
C *****
C
C SUBROUTINE DIBUJO(N)
C
C N = NUMERO DE PIRAMIDES A DIBUJAR.
C *****
C
C DIMENSION IOPT(4,20),AN(20)
C COMMON/A/DATOS(90,40),ICOD(20),NGE(20),IAG(20),IAGF(20)
C COMMON/B/ROTUL(80,21)
C COMMON/C/PARAM(3,20)
C COMMON/D/ESCAL,SEPAR,ESCA,SEPAR1,BASE,SEPARY
C INTEGER OUT
C CHARACTER ROTUL,RN,RESP
C CHARACTER*56 MENSA
C LOGICAL SUPER,COMP,ROT,LEY
C
C IOPT = MATRIZ DONDE SE ALMACENAN LOS OPTATIVOS DE DIBUJO QUE EX-
C PRESA EL USUARIO.
C AN = VECTOR DONDE SE ALMACENAN LAS FECHAS QUE SE CORRESPONDEN
C CON LAS PIRAMIDES.
C TODOS LOS ARRAYS QUE FORMAN PARTE DE LAS ZONAS COMUNES A Y B YA HAN
C SIDO DESCRITOS EN EL PROGRAMA PRINCIPAL.
C PARAM = MATRIZ DONDE VAN A ALMACENARSE LAS MODIFICACIONES A LOS PA-
C METROS STANDARD DE DIBUJO, QUE EL USUARIO SOLICITE.
C TODAS LAS VARIABLES QUE ESTAN INCLUIDAS EN LA ZONA COMUN D, SON VA-
C RIABLES INTERNAS DE DIBUJO.
C *****
C INICIALIZACION DE VARIABLES.
C
C DATA IN,OUT/7,6/
C
C LA INFORMACION REFERENTE A LAS MODIFICACIONES DE LOS PARAMETROS
C STANDARD DE DIBUJO, A LOS OPTATIVOS DE DIBUJO EXPRESADOS POR EL
C USUARIO Y A LAS FECHAS QUE SE CORRESPONDEN CON LAS PIRAMIDES FOR-
C MA PARTE DEL FICHERO FTOTFO01.
C
C DATA RMARG/1./
C DATA RN,RESP/'R',' '/
C DATA MENSA/'MONTAR OTRA HOJA,POR FAVOR.PARA CONTINUAR,RESPONDER'/'
C *'/'
C *****
C
C NM1=N+1
C
C LECTURA DE MODIFICACIONES DE PARAMETROS STANDARD.
C
C DO 1 I=1,5
C DO 2 J=1,20
C PARAM(I,J)=0.
C IF(1.E0,5) GOTO 2

```

```

      IOPT(I,J)=0
      2 CONTINUE
      1 CONTINUE
      READ(IN,100) (ROTUL(I,NM1),I=1,42),IPAR
100  FORMAT(42A1,T49,I2)
      WRITE(OUT,101) (ROTUL(I,NM1),I=1,42),IPAR
101  FORMAT(1X,42A1,T50,I2)
      IF(IPAR,EQ,1) GOTO 15
      DO 4 J=1,N
      IF(IPAR,EQ,2.AND.J.GT.1) GOTO 3
      READ(IN,102) (PARAM(I,J),I=1,5)
      WRITE(OUT,103) (PARAM(I,J),I=1,5)
102  FORMAT(5F10,0)
103  FORMAT(5F10,2)
      GOTO 4
      3 CONTINUE
      DO 5 I=1,5
      PARAM(I,J)=PARAM(I,J-1)
      5 CONTINUE
      4 CONTINUE
C
C*****
C
C  LECTURA DE OPTATIVOS Y DE FECHAS.
C
      15 CONTINUE
      READ(IN,200) (ROTUL(I,NM1),I=1,11),IOPTA
200  FORMAT(11A1,T19,I2)
      WRITE(OUT,201) (ROTUL(I,NM1),I=1,11),IOPTA
201  FORMAT(1X,11A1,T20,I2)
      IF(IOPTA,EQ,1) GOTO 35
      DO 24 J=1,N
      IF(IOPTA,EQ,2.AND.J.GT.1) GOTO 23
      READ(IN,202) (IOPT(I,J),I=1,4),ANC(J)
      WRITE(OUT,202) (IOPT(I,J),I=1,4),ANC(J)
202  FORMAT(T9,I2,T19,I2,T29,I2,T39,I2,T46,F5,0)
      GOTO 24
      23 CONTINUE
      DO 25 I=1,4
      IOPT(I,J)=IOPT(I,J-1)
      25 CONTINUE
      24 CONTINUE
C
C*****
C
C  35 CONTINUE
C
C  ASIGNACION INICIAL DE VALORES.
C
      YMAX= 81.
      XMAX= 130.
      CUADRO=((YMAX-1.)/4.)-1.
      X=0.
      Y=0.
      XSUM=0.
      YSUM=0.
      GRANX=1.
      RNGRF=1.

```

```

DESP=0.
C
C SE ABRE EL DIBUJO POR PRIMERA VEZ.
C
C CALL PLOTS(0,0,6)
C
C*****
C
C EN CADA ITERACION SE CONSIDERA UNA PIRAMIDE DISTINTA.
C
C DO 6 J=1,N
C*****
C
C XP=0.
C YP=0.
C IF(PARAM(1,J),NE.0.)CALL FACTOR(PARAM(1,J))
C CUADRX=CUADRO
C J2=J*2
C J2M1=J2-1
C COMP=.FALSE.
C SUPER=.FALSE.
C
C*****
C
C DIBUJO DE LAS PIRAMIDES.
C
C IF(ICDD(J).EQ.1) GOTO 7
C COMP=.TRUE.
C IF(ICDD(J).EQ.2) GOTO 8
C SUPER=.TRUE.
C CALL PIRAM(CUADRG, CUADRO, RMARG, XP, YP, DATOS(1,
C *2*IPIR-1), DATOS(1,2*IPIR), NGE(IPIR), IAGF(IPIR), .FALSE., J)
C CALL PIRAM(CUADRO, CUADRO, RMARG, XP, YP, DATOS(1, J2M1), DATOS(1, J2),
C *NGE(J), IAGF(J), SUPER, J)
C GOTO 9
C 7 CONTINUE
C IPIR=J
C CALL PIRAM(CUADRO, CUADRO, RMARG, XP, YP, DATOS(1, J2M1), DATOS(1, J2),
C *NGE(J), IAGF(J), SUPER, J)
C GOTO 9
C 8 CONTINUE
C IF(ICDD(J-1).NE.2)
C *CALL PIRAM(CUADRO, CUADRO, RMARG, XP, YP, DATOS(1,
C *2*IPIR-1), DATOS(1,2*IPIR), NGE(IPIR), IAGF(IPIR), .FALSE., J)
C IF(ICDD(J-1).EQ.2) DESP=0.5
C CALL PIRAM2(8, DESP, DATOS(1, J2M1), DATOS(1, J2), NGE(J), IAGF(J), J)
C DESP=0.
C 9 CONTINUE
C
C*****
C
C ROT=.FALSE.
C LEY=.FALSE.
C
C*****
C
C DIBUJO DE ROTULOS Y LEYENDAS.

```

```

C
  IF(IOPT(1,J),NE.0)CALL ROTULO(CUADRO,J,ROT)
  IF(IOPT(4,J),NE.0)CALL ANNAC(J,AN(J))
  IF(IOPT(2,J),NE.0)CALL LEYEN1(CUADRO,LEY,DX,IOPT(2,J))
  IF(IOPT(3,J),NE.0)CALL LEYEN2(CUADRO,LEY,DX)
  IF(ROT) GOTO 10
  CALL NUMBER(1.,(CUADRO-1.),1.,RNGRF,0.,-1)
C
C+*****
C
  10 CONTINUE
  IF(LEY) CUADRX=CUADRX+DX
C
C+*****
C
C CONTROL DE LA PLUMA PARA SITUARLA ADECUADAMENTE EN EL PROXIMO DI-
C BUJO.
C
  IF(SUPER) GOTO 11
  IF(ICOD(J+1),E0.1) GOTO 11
  IF(ICOD(J),NE.ICOD(J+1)) GOTO 11
  GOTO 6
C
C SE VA A HACER OTRO DIBUJO.
C
  11 CONTINUE
  CALL WHERE(VX,VY,FCT)
C
C EL ANTERIOR DIBUJO SALE ENMARCADO POR (CUADRX*FCT,CUADRO*FCT).
C
  XTDP=CUADRX*FCT
  IF(GRANX,LT,XTDP)GRANX=XTDP
  YTOP=CUADRO*FCT
  CALL FACTOR(1.)
  XMARC=0.
  YMARC=0.
  CALL MARCO(XMARC,YMARC,XTDP,YTOP)
C
C AHORA SE MUEVE LA PLUMA PARA EL SIGUIENTE DIBUJO.
C
  RNGRF=RNGRF+1.
  Y=YTOP+1.
  YSUM=YSUM+Y
  IF((YSUM+Y).GT.YMAX) GOTO 12
  CALL PLOT(0.,Y,-3)
  GOTO 6
  12 CONTINUE
  X=GRANX+2.
  XSUM=XSUM+X
  IF((XSUM+X).GT.XMAX) GOTO 14
  Y=-YSUM-Y
  CALL PLOT(X,Y,-3)
  YSUM=0.
  GRANX=1.
  GOTO 6
C
C ESTE CONJUNTO DE INSTRUCCIONES CAUSAN QUE SE INTERRUMPA EL PROGRA-
C MA HASTA QUE ESTE MONTADA UNA NUEVA HOJA EN EL PLOTTER.

```

```
C
14 CONTINUE
DO 40 NIN=1,100
CALL PLOT(0.,0.,3)
CALL PLOT(0.,1.,3)
40 CALL PLOT(0.,0.,3)
CALL SEND(RN,MENSA,RESP)
X=0.
Y=0.
XSUM=0.
YSUM=0.
GRANX=1.
C
C*****
C*****
C
6 CONTINUE
C
C*****
C
SE CIERRA DEFINITIVAMENTE EL DIBUJO.
CALL PLOT(-1.,-1.,999)
C
C*****
C
RETURN
END
```



```
C
C *****
C * SUBROUTINE QUE ALMACENA EN LA ZONA COMUN E, LOS VALDRES DE LOS *
C *VERTICES DE LOS RECTANGULOS QUE SE TRAMAN EN LAS PIRAMIDES SUPER- *
C *PUESTAS, PARA FACILITAR SU INTERPRETACION. *
C *****
C
SUBROUTINE DIFER(X,Y,SUPER,ISW,J)
COMMON/E/XS(4,180),YS(4,180)
LOGICAL SUPER
GOTO(1,2),ISW
1 CONTINUE
I=1
IF(SUPER)I=4
XS(I,J)=X
YS(I,J)=Y
RETURN
2 CONTINUE
I=2
IF(SUPER)I=3
XS(I,J)=X
YS(I,J)=Y
C
C *****
C
RETURN
END
```

```
C
C *****
C *   ESTA SUBROUTINA GIRA EL PUNTO (X,Y) UN ANGULO +W = (XN,YN)   *
C *****
C
C   SUBROUTINE G1(XN,YN,X,Y,W)
C   CW=COS(W*3.141593/180.)
C   SW=SIN(W*3.141593/180.)
C   XN=CW*X-SW*Y
C   YN=SW*X+CW*Y
C
C *****
C
C   RETURN
C   END
```

```
C
C *****
C *   ESTA SUBROUTINA GIRA EL PUNTO (X,Y) UN ANGULO -W = (XN,YN)   *
C *****
C
C   SUBROUTINE G2(XN,YN,X,Y,W)
C   CW=COS(W*3.141593/180.)
C   SW=SIN(W*3.141593/180.)
C   XN=CW*X+SW*Y
C   YN=-SW*X+CW*Y
C *****
C
C   RETURN
C   END
```

```

C
C *****
C * EN ESTA SUBROUTINA SE LEEN LOS PESOS DE LA VARIABLE, O VARIABLES*
C *(CLUSTER) -NUNCA MAS DE 10, TAL COMO ESTÁ DISEÑADO EL PROGRAMA,*
C *QUE AFECTAN A CADA CASO, Y LOS OPTATIVOS QUE PUEDE DEFINIR EL *
C *USUARIO PARA ELEGIR ENTRE LOS DIVERSOS PROCEDIMIENTOS DE CLASI- *
C *FICACION QUE SE OFERTAN, Y PARA DECIDIR EL NUMERO DE INTERVALOS *
C *DE CLASIFICACION -ENTRE 2 Y 10-, ASI COMO LA NATURALEZA DE LAS *
C *TRAMAS ASOCIADAS A ESTOS.*
C *****
C
C SUBROUTINE INPUT(N,IOCLAS)
C
C     N = NUMERO DE CASOS.
C     IOCLAS = OPTATIVO DE PROCEDIMIENTO DE CLASIFICACION.
C *****
C
C     COMMON/A/ PESO(100)
C     COMMON/C/ PESOS(100,10)
C     COMMON/D/ ICLAS,ITRAMA(10)
C     COMMON/E/ UMBR(9)
C     CHARACTER FMT(80)
C     INTEGER OUT
C
C     PESO = VECTOR DONDE SE ALMACENAN LOS PESOS DE LA VARIABLE SOBRE
C     LAS AREAS EN CONSIDERACION.
C     PESOS = MATRIZ DONDE SE ALMACENAN LOS PESOS DE DIVERSAS VARIABLES
C     (NUNCA MAS DE 10) SOBRE LAS AREAS EN CONSIDERACION.
C     ICLAS = NUMERO DE INTERVALOS DE CLASIFICACION.
C     ITRAMA = VECTOR ENTERO DONDE SE ALMACENA, MEDIANTE CODIGOS, LA NA-
C     TURALEZA DE LAS TRAMAS A REPRODUCIR.
C     UMBR = VECTOR DE UMBRALES DE CLASIFICACION.
C     FMT = VECTOR DE CARACTERES PARA LA DESCRIPCION DEL FORMATO DE
C     LOS DATOS QUE APORTA EL USUARIO.
C *****
C
C     INICIALIZACION DE LAS VARIABLES DE LECTURA Y ESCRITURA.
C
C     DATA IN,OUT/5,6/
C *****
C
C     SE LEE AHORA EL OPTATIVO (ISTAND) QUE DEFINE SI SE VA A UTILIZAR
C     EL PROCEDIMIENTO ABSOLUTAMENTE STANDARD (ISTAND=1), O SI SE VA A
C     MANIFESTAR ALGUNA PETICION ESPECIAL(ISTAND=2).
C
C     READ(IN,100) ISTAND
C     100 FORMAT(I1)
C     IF(ISTAND.NE.1) GOTO 1
C *****
C
C     SE LEE AHORA EL OPTATIVO DE DESCRIPCION DE FORMATO.
C     IFMT: 1. SE LEEN LOS DATOS CON FORMATO STANDARD.
C     IFMT: 2. SE LEEN CON EL FORMATO QUE DECIDE EL USUARIO.
C

```

```

      READ(IN,103) IFMT
      IF(IFMT.EQ.2) GOTO 13
      IN=7
      READ(IN,101) N,(PESO(I),I=1,N)
101  FORMAT(13/(8F10.0))
      GOTO 14
13  CONTINUE
      READ(IN,104) (FMT(I),I=1,80)
      IN=7
      READ(IN,FMT) N,(PESO(I),I=1,N)
14  WRITE(OUT,102) N,(PESO(I),I=1,N)
102  FORMAT(5X,13/(5X,8F10.3))
C
C*****
      CALL CLASST(N)
      RETURN
C*****
C
C   SE HA DECIDIDO NO UTILIZAR LA VERSION STANDARD.
C
C   1 CONTINUE
C
C   SE LEEN AHORA LOS SIGUIENTES OPTATIVOS:
C   IOCLAS: 1. CLASIFICACION SEGUN UMBRALES QUE APORTA EL USUARIO.
C           2. CLASIFICACION POR CENTILES.
C           3. CLASIFICACION POR UNIDADES DE DESVIACION STANDARD.
C           4. CLUSTER.
C           5. CLUSTER, A PARTIR DE MATRIZ DE DISTANCIAS.
C   ICLAS: NUMERO DE INTERVALOS(NUNCA MAYOR QUE 10).
C   IFMT: 1. SE LEEN LOS DATOS CON FORMATO STANDARD.
C         2. SE LEEN LOS DATOS CON EL FORMATO QUE DECIDE EL USUARIO.
C   ITRANS: 1. SE LEEN LOS DATOS EN INPUT.
C           2. SE LEEN LOS DATOS EN TRANS1 O EN TRANS2 (ESTO PERMITE
C             LECTURAS ESPECIALES Y TRANSFORMAR LOS PESOS DE LA VARIABLE,
C             SI SE DEFINEN ADECUADAMENTE LAS SUBROUTINAS TRANS1 O TRANS2).
C   RIDT: 1. EN LA CLASIFICACION POR UNIDADES DE DESVIACION STANDARD
C          LA AMPLITUD DE LAS CLASES ES DE UNA DESVIACION STANDARD.
C          2. EN LA CLASIFICACION POR UNIDADES DE DESVIACION STANDARD
C          LA AMPLITUD DE LAS CLASES ES DE 1/2 DESVIACION STANDARD.
C
      READ(IN,203) IOCLAS,ICLAS,IFMT,ITRANS,RIDT
      WRITE(OUT,203) IOCLAS,ICLAS,IFMT,ITRANS,RIDT
203  FORMAT(4I5,F5.0)
C
C   LECTURA DE LAS TRAMAS ASOCIADAS A CADA INTERVALO.
C
      READ(IN,103) (ITRAMA(I),I=1,ICLAS)
      WRITE(OUT,103) (ITRAMA(I),I=1,ICLAS)
103  FORMAT(16I5)
C
C
      IF(IOCLAS.EQ.4.OR.IOCLAS.EQ.5) GOTO 2
      IF(ITRANS.EQ.2) GOTO 3
      IF(IFMT.EQ.2) GOTO 4
C
C   LECTURA STANDARD
C
      IN=7

```

```

      READ(IN,101) N,(PESO(I),I=1,N)
      WRITE(OUT,102) N,(PESO(I),I=1,N)
      GOTO 5
C
C   4 CONTINUE
C
C   LECTURA CON EL FORMATO QUE APORTA EL USUARIO.
C
      READ(IN,104) (FMT(I),I=1,80)
104  FORMAT(80A1)
      IN=7
      READ(IN,FMT) N,(PESO(I),I=1,N)
      WRITE(OUT,102) N,(PESO(I),I=1,N)
      GOTO 5
C
C   3 CONTINUE
C
C   LECTURA POR TRANS1.
C
      READ(IN,108) N
      CALL TRANS1(N)
C
C   5 CONTINUE
C
C   EMPIEZA LA LLAMADA A LOS DIFERENTES SUBPROGRAMAS DE CLASIFICACION.
C
      IF(10CLAS.NE.2) GOTO 6
C*****
      CALL CLASCT(N)
      RETURN
C*****
C
C   6 CONTINUE
      IF(10CLAS.NE.3) GOTO 7
C*****
      CALL CLASDT(N,R1DT)
      RETURN
C*****
C
C   7 CONTINUE
      IN=5
      READ(IN,105)(UMBR(I),I=1,CLAS-1)
105  FORMAT(10F8.0)
C*****
      CALL CLASUM(N)
      RETURN
C*****
C
C   2 CONTINUE
C
C   SE RECURRE A UN PROCEDIMIENTO CLUSTER.
C
      READ(IN,108) N,NVAR
108  FORMAT(1X,2I3)
C

```

```

C      IF(I*TRANS.EQ.2) GOTO 8
C      IF(IFMT.EQ.2) GOTO 9
C      LECTURA STANDARD.
C      IN=7
C      DO 10 I=1,N
C      READ(IN,106) (PESOS(I,J),J=1,NVAR)
C      WRITE(OUT,107) (PESOS(I,J),J=1,NVAR)
10 CONTINUE
106 FORMAT(8F10.0)
107 FORMAT(5X,8F10.3)
C      GOTO 12
C      9 CONTINUE
C      LECTURA CON EL FORMATO QUE DEFINE EL USUARIO.
C      READ(IN,104)(FMT(I),I=1,80)
C      IN=7
C      DO 11 I=1,N
C      READ(IN,FMT) (PESOS(I,J),J=1,NVAR)
C      WRITE(OUT,107) (PESOS(I,J),J=1,NVAR)
11 CONTINUE
C      GOTO 12
C      8 CONTINUE
C      LECTURA POR TRANS2.
C      CALL TRANS2(N,NVAR)
C      12 CONTINUE
C      CALL CLUSTR(N,NVAR,IOCLAS)
C      RETURN
C      .....
C      END

```

```

C
C *****
C *SUBROUTINA DE INTERPOLACION TEMPORAL LINEAL.*
C *****
C
C   SUBROUTINE ITEMP(P,MAX,N,T1,T2,NI,TAM)
C
C   P   = PESOS.
C   MAX = DIMENSION DE P EN EL PROGRAMA PRINCIPAL.
C   N   = NUMERO DE CASOS.
C   T1  = FECHA INICIAL.
C   T2  = FECHA FINAL.
C   NI  = NUMERO DE IMAGENES ENTRE T1 Y T2.
C   TAM = TAMANO DE LOS CARACTERES DE LA FECHA.
C
C   EN EL PROGRAMA PRINCIPAL SE HAN LEIDO LOS PESOS EN EL MOMENTO T1,
C   EN P(1,1), Y EN EL MOMENTO T2 EN P(1,2). TAMBIEN SE HAN LEIDO T1,T2
C   NI Y TAM.
C *****
C
C   DIMENSION P(MAX,3)
C   DIMENSION TAN(400)
C
C   EN TAN(I) SE VAN A ALMACENAR LAS PENDIENTES DE TODAS LAS LINEAS DE
C   INTERPOLACION ENTRE P(1,1) Y P(1,2).
C *****
C
C   SE CALCULAN LOS VALORES DE DICHAS PENDIENTES.
C
C   DO 1 I=1,N
C     TAN(I)=(P(1,2)-P(1,1))/(T2-T1)
C 1 CONTINUE
C
C   TINC = INTERVALO TEMPORAL ENTRE DOS IMAGENES.
C
C     TINC=(T2-T1)/FLOAT(NI)
C
C   CALCULO DE CADA IMAGEN.
C
C     T=T1
C     DO 2 IT=1,NI
C       IF(IT.EQ.1) GOTO 3
C     DO 5 I=1,N
C       P(I,3)=P(I,1)+(TAN(I))*(TINC*(IT-1))
C 5 CONTINUE
C
C *****
C     CALL BLOCKD(P(1,3),MAX,N,T,TAM)
C *****
C
C     GOTO 4
C 3 CONTINUE
C
C *****
C     CALL BLOCKD(P(1,1),MAX,N,T,TAM)
C *****

```



```
C 4 CONTINUE  
  T=T+TINC  
  2 CONTINUE  
  RETURN
```

```
C .....  
C
```

```
  END
```

```

C
C *****
C *   ESTA SUBROUTINA (LLAMADA DESDE CACHE), DEVUELVE EN (X,Y) LAS   *
C * COORDENADAS DE UN PUNTO DE INTERSECCION ENTRE LA NUEVA LINEA Y LA *
C * ENVOLVENTE. *
C *****
C
C   SUBROUTINE INTER(X,Y,C,D,ISW)
C   DIMENSION C(3),D(3)
C   X1=C(1)
C   X2=D(1)
C   THR=0.0001
C   Y1=C(2)
C   Y2=D(2)
C   YP1=C(3)
C   YP2=D(3)
C   IF(ABS(X2-X1).GT.THR) GO TO 1
C   X=X1
C   GOTO(11,12),ISW
C11 Y=AMIN1(YP2,Y1)
C *****
C   RETURN
C *****
C12 Y=AMIN1(YP1,Y2)
C *****
C   RETURN
C *****
C1 CONTINUE
C   H=X2-X1
C   B1=YP1-Y1
C   B2=Y2-YP2
C   X=H/2
C   IF(B1+B2.NE.0.) X=H*B1/(B1+B2)
C   X=AMIN1(H,AMAX1(0.,X))
C   X=X1+X
C   Y=Y1+(Y2-Y1)*(X-X1)/(X2-X1)
C   RETURN
C *****
C
C   END

```

```
C
C *****
C *   ESTA SUBROUTINA CALCULA LAS COORDENADAS DEL PUNTO DE CORTE DE   *
C * DOS RECTAS, DE UNA DE LAS CUALES SE CONOCE LA PENDIENTE Y UN PUN- *
C * TO POR EL QUE PASA, Y DE LA OTRA DOS PUNTOS POR LOS QUE PASA.   *
C *****
C
C   SUBROUTINE INTERS(X,Y,PEND,XR1,YR1,XR2,YR2,XC,YC)
C   C=Y-PEND*X
C   B=(YR1-YR2)/(XR1-XR2)
C   A=(YR2*XR1-YR1*XR2)/(XR1-XR2)
C   XC=(C-A)/(B-PEND)
C   YC=C-PEND*XC
C *****
C
C   RETURN
C   END
```

```
C
C *****
C *  ESTA SUBROUTINA CALCULA LA COORDENADA Y DEL PUNTO DE COORDENADA *
C *X PERTENECIENTE A UNA RECTA QUE PASA POR LOS PUNTOS (X1,Y1) Y *
C *(X2,Y2). *
C *****
C
C   SUBROUTINE INTP(X,Y,X1,Y1,X2,Y2)
C     Y=Y1
C *****
C     IF(X1.EQ.X2) RETURN
C *****
C     Y=Y1+(X-X1)*(Y2-Y1)/(X2-X1)
C     RETURN
C *****
C
C   END
```

```
C
C *****
C *   ESTA SUBROUTINA CAMBIA EL SENTIDO DE DESCRIPCION, POR SUCESSION*
C *DE PUNTOS, DE UNA LINEA.*
C *****
C
  SUBROUTINE INVERT(LINEA,L)
  REAL LINEA(L),A(200)
  DO 1 I=1,L,2
    IM1=I+1
    K=L-(I-1)
    KM1=K-1
    A(K)=LINEA(IM1)
    A(KM1)=LINEA(I)
  1 CONTINUE
  DO 2 I=1,L
    LINEA(I)=A(I)
  2 CONTINUE
  RETURN
C
C *****
C
  END
```

```

C
C *****
C *   ESTA SUBROUTINA SITUA LA LEYENDA DE LAS TRAMAS UTILIZADAS, SOBRE
C *   UN RECTANGULO DE DIMENSIONES XL*YL, CUYO ANGULO INFERIOR IZQUIERDO
C *   SE SITUA EN (XD,YD).
C *   ESTA SUBROUTINA NO ALTERA EL ORIGEN DE COORDENADAS DEFINIDO
C *   ANTES DE LLAMARLA.
C *****
C
C   SUBROUTINE LEGEND(XD,YD,XL,YL)
C *****
C
C   COMMON/D/ICLAS,IT(10)
C   COMMON/F/IFR(10,2),UMBIS(10,2)
C   DIMENSION XYS(4,2)
C *****
C
C   SE TRASLADA EL ORIGEN DE COORDENADAS A (XD,YD).
C
C   CALL PLOT(XD,YD,-3)
C *****
C
C   *****
C   *SE DEFINEN LOS PARAMETROS FUNDAMENTALES PARA LA ROTULACION.*
C   *****
C
C   MARGEN GENERAL RESPECTO DEL CUADRO.
C
C   XYM=0.5
C
C   TAMAÑO DE UNA LINEA DE ROTULACION.
C
C   YVAR=(YL-2*XYM)/(FLOAT(ICLAS)+3.)
C
C   TAMAÑO DE LAS LETRAS DE LA PALABRA LEYENDA Y COORDENADA X E Y DEL
C   PUNTO A PARTIR DEL CUAL DEBE SITUARSE EL ROTULO.
C
C   TAMLEY=AMINI(0.5,YVAR)
C   ORXLEY=(XL-7*TAMLEY)/2.
C   ORYLEY=(YVAR-TAMLEY)/2.
C
C   LADO DEL CUADRADO QUE VA A RECIBIR LA TRAMA EN LA LEYENDA Y TAMAÑO
C   DE LOS NUMEROS Y CARACTERES QUE LA EXPLICAN.
C
C   TAMNUM=(XL-2*XYM-1.)/30.
C   TAMNUM=AMINI(TAMNUM,YVAR/3.)
C
C
C   ROTULACION.
C
C   ORXLRT=(XL-TAMNUM*30-1.)/2.
C *****
C *****
C

```



```

C SE ROTULA ' < VAR <'
C
4 STARTX=ORXLRT+1,+TAMNUM*13
  IF(J.E0.1)
  *CALL SYMBOL(STARTX,YBASE,TAMNUM,' VAR< ',0.,7)
  .IF((J.NE.1).AND.(J.NE.ICLAS))
  *CALL SYMBOL(STARTX,YBASE,TAMNUM,' <VAR< ',0.,7)
  IF(J.E0.ICLAS)
  *CALL SYMBOL(STARTX,YBASE,TAMNUM,' <VAR ',0.,7)
C
  IF(J.E0.ICLAS) GOTO 1
C
C SE INVESTIGA EL VALOR DEL UMBRAL SUPERIOR PARA QUE TODOS LOS
C UMBRALES APAREZCAN AJUSTADOS A LA DERECHA.
C
  STARTX=ORXLRT+1,+TAMNUM*20
  DO 5 IUM=1,6
  IUMA=7-IUM
  IF(ABS(UMBIS(J,2)).GE.10.**IUMA) GOTO 5
  STARTX=STARTX+TAMNUM
5 CONTINUE
  IF(UMBIS(J,2).LT.0.)STARTX=STARTX-TAMNUM
C
C SE ROTULA EL UMBRAL SUPERIOR.
C
  CALL NUMBER(STARTX,YBASE,TAMNUM,UMBIS(J,2),0.,2)
C
C*****
C
  1 CONTINUE
C
C*****
C
C SE DEVUELVE EL ORIGEN DE COORDENADAS AL PUNTO EN QUE SE ENCONTRABA
C ANTES DE LLAMAR A ESTA SUBROUTINA.
C
  CALL PLOT(-X0,-Y0,-3)
  RETURN
C
C*****
C
  END

```



```

C
C *****
C *   ESTA SUBROUTINA SITUA A LA DERECHA DE LA PIRAMIDE LA LEYENDA DE *
C * LAS TRAMAS QUE SE UTILIZAN EN EL DIBUJO DE LAS PIRAMIDES COMPUES- *
C * TAS. *
C *****
C
SUBROUTINE LEYENI(RL,LEY,DX,NUM)
COMMON/E/XS(4,180),YS(4,180)
COMMON/F/SALTO
LOGICAL LEY
LEY=.TRUE.
CALL PLOT(RL,0.,-3)
TAM=0.3
XN=1.
IL=4
IF(NUM.EQ.2)IL=6
RIL=FLOAT(IL)
DO 1 I=1,IL
RI=1.
IF(IL.EQ.4.AND.(I.EQ.2.OR.I.EQ.4))RI=2.
IF(IL.EQ.6.AND.(I.EQ.2.OR.I.EQ.5))RI=2.
IF(IL.EQ.6.AND.(I.EQ.3.OR.I.EQ.6))RI=3.
YN=RL-RL/RIL*(I-1)-(RL/(2.*RIL)-TAM/2.)
CALL NUMBER(XN,YN,TAM,RI,0.,-1)
CALL SYMBOL(999.,999.,TAM,'-',0.,1)
IF(IL.EQ.6) GOTD 3
GOTD(2,2,1,1),I
2 CONTINUE
CALL WHERE(XC,YC,R)
YC=YC+TAM/2.
XC=XC+0.1
RCX=RL/6.-2.
YC=YC-(RCX-0.2)/2.
RCX=RCX-0.2
CALL CUADRA(XC,YC,RCX)
IF(I.EQ.2) CALL TRAMEL(1,4,0.,SALTO,1)
GOTO 1
3 CONTINUE
GOTD(4,4,4,1,1),I
4 CONTINUE
CALL WHERE(XC,YC,R)
YC=YC+TAM/2.
XC=XC+0.1
RCX=RL/6.-2.
YC=YC-(RCX-0.2)/2.
RCX=RCX-0.2
CALL CUADRA(XC,YC,RCX)
IF(I.EQ.2) CALL TRAMEL(1,4,0.,SALTO,1)
IF(I.EQ.3) CALL TRAMEL(1,4,0.,SALTO/2.,1)
1 CONTINUE
CALL PLOT(-RL,0.,-3)
DX=RL/3.
C
C *****
C
RETURN
END

```

```

C
C *****
C *   ESTA SUBROUTINA SITUA, A LA DERECHA DE LA PIRAMIDE, UNA LEYENDA *
C *DE LAS TRAMAS QUE SE UTILIZAN PARA ILUSTRAR LA SUPERPOSICION DE *
C *DOS PIRAMIDES. LA TRAMA MENOS DENSA SIGNIFICA SIEMPRE UN MAYOR DE- *
C *SARROLLO DE LA COHORTE CORRESPONDIENTE A LA PIRAMIDE DE REFERENCIA*
C *****
C
SUBROUTINE LEYEN2(RL,LEY,DX)
COMMON/E/XS(4,180),YS(4,180)
LOGICAL LEY
LEY=.TRUE.
CALL PLDT(RL,0.,-3)
TAM=0.3
XN=1.
DO 1 I=1,4
RI=1.
IF(I.EQ.2.OR.I.EQ.4)RI=2.
YN=RL-RL/4.*(I-1)-(RL/8.-TAM/2.)
CALL NUMBER(XN,YN,TAM,RI,0.,-1)
CALL SYMBOL(999.,999.,TAM,'-',0.,1)
GOTO(2,2,1,1),I
2 CONTINUE
CALL WHERE(XC,YC,R)
YC=YC+TAM/2.
XC=XC+0.1
RCX=RL/6.-2.
YC=YC-(RCX-0.2)/2.
RCX=RCX-0.2
CALL CUADRA(XC,YC,RCX)
IF(I.EQ.1) CALL TRAMEL(1,4,90.,0.02,1)
IF(I.EQ.2) CALL TRAMEL(1,4,90.,0.07,1)
1 CONTINUE
CALL PLDT(-RL,0.,-3)
DX=RL/3.
C
C *****
C
RETURN
END

```

```
C
C *****
C *   ESTA SUBROUTINA DIBUJA UN RECTANGULO DE XD * YD, CUYO VERTICE *
C * INFERIOR IZQUIERDO SE SITUA EN (X,Y). *
C *****
C
  SUBROUTINE MARCO(X,Y,XD,YD)
    CALL PLOT(X,Y,3)
    DO 1 I=1,4
      GOTO(2,3,4,5),I
    2 X=X+XD
      GOTO 6
    3 Y=Y+YD
      GOTO 6
    4 X=X-XD
      GOTO 6
    5 Y=Y-YD
    6 CONTINUE
    CALL PLOT(X,Y,2)
    1 CONTINUE
C
C *****
C
  RETURN
  END
```

```

C
C *****
C * ESTA SUBROUTINA (LLAMADA DESDE CACHE) ALMACENA EN W TODOS LOS PUN*
C *TOS DE LA NUEVA LINEA Y LOS DE LA ENVOLVENTE ORDENADOS POR SU COOR*
C *DENADA X. ADEMÁS, CALCULA, RECORRIENDO A INTP, EL VALOR DE LA EN- *
C *VOLVENTE EN CADA PUNTO DE LA NUEVA LINEA Y VICEVERSA, POR INTERPO- *
C *LACION LINEAL. ESTA ULTIMA INFORMACION TAMBIEN SE ALMACENA EN W. *
C *****
C
SUBROUTINE MERGE(W,LW,NPM,V,M,C,N,KIN,KFN)
LOGICAL LW
DIMENSION W(3,1),LW(1),V(2,M),C(2,N)
NPM=N+M
IF(V(1,1).LT.C(1,1)) GO TO 10
XP1=C(1,1)
YP1=C(2,1)
GO TO 20
10 CONTINUE
X1=V(1,1)
Y1=V(2,1)
20 CONTINUE
K=0
J=0
I=1
KIN=1
1 CONTINUE
J=J+1
IF(J.GT.N) GO TO 5
2 CONTINUE
IF(V(1,I).LT.C(1,J)) GO TO 3
K=K+1
W(1,K)=C(1,J)
W(3,K)=C(2,J)
LW(K)=.FALSE.
IF(J.EQ.1) KIN=K
CALL INTP(C(1,J),W(2,K),XP1,YP1,V(1,I),V(2,I))
X1=C(1,J)
Y1=C(2,J)
GO TO 1
3 CONTINUE
K=K+1
W(1,K)=V(1,I)
W(2,K)=V(2,I)
LW(K)=.TRUE.
CALL INTP(V(1,I),W(3,K),X1,Y1,C(1,J),C(2,J))
XP1=V(1,I)
YP1=V(2,I)
I=I+1
IF(I.LE.M) GO TO 2
IF(J.EQ.1) KIN=K+1
DO 4 JJ=J,N
K=K+1
W(1,K)=C(1,JJ)
W(3,K)=C(2,JJ)
CALL INTP(C(1,JJ),W(2,K),XP1,YP1,C(1,N),C(2,N))
LW(K)=.FALSE.
4 CONTINUE
NPM=K

```

```
      KFN=K
C*****
      RETURN
C*****
      5 CONTINUE
      NPM=K
      KFN=K
      DO 6 I1=1,M
      K=K+1
      W(1,K)=V(1,I1)
      W(2,K)=V(2,I1)
      CALL INTP(V(1,I1),W(3,K),X1,Y1,V(1,M),V(2,M))
      LW(K)=.TRUE.
      6 CONTINUE
      NPM=K
      RETURN
C
C*****
C
      END
```

```

C
C *****
C * ESTA SUBROUTINA SITUA SOBRE EL PUNTO DE COORDENADAS (X,Y) UNA *
C *MOTA CIRCULAR DE RADIO P. *
C *****
C
SUBROUTINE MOTA (X,Y,P)
DIMENSION O(25),R(25)
DATA N/0/
IF(N.NE.0) GO TO 2
N=24
O(1)=1.
R(1)=0.
DO 1 I=1,N
W=2.*I*3.141593/N
O(I+1)=COS(W)
R(I+1)=SIN(W)
1 CONTINUE
2 CONTINUE
D=0.02
M=MAXO(1,INT(P/D))
T=P
DO 4 I=1,M
CALL PLOT(T*O(1)+X,T*R(1)+Y,3)
DO 3 J=1,N
CALL PLOT(T*O(J+1)+X,T*R(J+1)+Y,2)
3 CONTINUE
T=T-D
4 CONTINUE
C
C *****
C
RETURN
END

```

```
C
C *****
C *   ESTA SUBROUTINA MODIFICA EL ORDEN DE DESCRIPCION DE LOS PUNTOS *
C * COMUNES DEL POLIGONO A FUNDIR, CUANDO LA FRONTERA EN LA ENVOLVEN- *
C * TE *PASA POR O*. *
C *****
C
C   SUBROUTINE NEVA1
C
C   COMMON/H/IE(1000),IA(50)
C   COMMON/GIRIA/NGIR,NNJD
C
C   DO 1 I=1,NGIR
C   IAUX=IA(NNJD)
C   DO 2 J=1,NNJD-1
C   IJ=NNJD-(J-1)
C   IJM1=IJ-1
C   IA(IJ)=IA(IJM1)
C 2 CONTINUE
C   IA(1)=IAUX
C 1 CONTINUE
C   RETURN
C
C *****
C
C   END
```

```

C
C *****
C *   ESTA SUBROUTINA ALMACENA EN LA MATRIZ A LAS COORDENADAS DE LOS *
C * PUNTOS QUE CONSTITUYEN EL CAMINO COMPLEMENTARIO POR EL GRAFO DEL *
C * POLIGONO A FUNDIR, QUE EMPIEZA Y TERMINA EN LOS MISMOS VERTICES *
C * QUE LA FRONTERA COMUN DE ESTE CON LA ENVOLVENTE. *
C *****
C
C   SUBROUTINE HENWAY(IP,I1,I2,I)
C
C   IP=NUMERO DEL POLIGONO A FUNDIR.
C   I1=NUMERO DE ORDEN EN LA LISTA QUE SE CORRESPONDE CON EL PRIMERO
C   DE LA SECCION DEL POLIGONO A REESCRIBIR.
C   I2=NUMERO DE ORDEN EN LA LISTA QUE SE CORRESPONDE CON EL ULTIMO
C   DE LA SECCION DEL POLIGONO A REESCRIBIR.
C   I=NUMERO DE PUNTOS QUE FORMAN EL CAMINO COMPLEMENTARIO.
C
C   COMMON/F/P(100,100),E(1000,2),A(50,2),DISMAX
C   COMMON/G/LP(50),M(50),NP,LE,INFOR,DUT
C   COMMON/L/RESTO
C
C   LOGICAL RESTO
C   INTEGER DUT,P,E,A,DISMAX
C
C   MAX=LP(IP)
C   IP2=IP*2
C   IP2M1=IP2-1
C
C   TEST PARA DECIDIR POR EL TIPO DE TRATAMIENTO NECESARIO.
C
C   IF(I2-I1)1,4,5
C
C   1 CONTINUE
C   IF(RESTO) GOTO 6
C
C   LOS VERTICES SON DECRECIENTES.
C
C   2 J=I1
C   I=0
C   3 CONTINUE
C   IF(J.EQ.MAX.AND.I2.EQ.1)RETURN *****
C   IF(J.EQ.(I2-1))RETURN *****
C
C   J=J+1
C   IF(J.GT.MAX)J=1
C   I=I+1
C   A(I,1)=P(J,IP2M1)
C   A(I,2)=P(J,IP2)
C   GOTO 3
C
C   4 CONTINUE
C
C   WRITE(OUT,100)
C   CALL EXIT
C
C   5 CONTINUE
C   IF(RESTO) GOTO 2

```



```
C
C  LOS VERTICES SON CRECIENTES.
C
6  J=1
   I=0
7  CONTINUE
   IF(J.EQ.1.AND.I2.EQ.MAX)RETURN
C
   IF(J.EQ.(I2+1))RETURN *****
C
   J=J-1
   IF(J.EQ.0)J=MAX
   I=I+1
   A(I,1)=P(J,IP2M1)
   A(I,2)=P(J,IP2)
   GOTO 7
C
C *****
C
100 FORMAT(' HAY PROBLEMAS EN NEWWAY')
END
```

```

C
C .....
C *   ESTA SUBROUTINA ORDENA LAS FILAS DE UNA MATRIZ.   *
C .....
C
SUBROUTINE ORDEN(A,MAX,M,N,KMAX)
DIMENSION A(MAX,N)
M1=M-1
IF(M1.LE.0)GOTO 5
DO 4 I1=1,M1
I=I1
I1=I+1
DO 2 J=I1,M

C
C   COMPARAR A(I,*) CON A(J,*)
C
DO 1 K=1,KMAX
IF(A(I,K).EQ.A(J,K))GOTO 1
IF(A(I,K).LT.A(J,K))GOTO 2
I=J
GOTO 2
1 CONTINUE
2 CONTINUE
IF(I.EQ.I1) GOTO 4

C
C   SUSTITUIR A(I,*) POR A(I1,*) Y VICEVERSA
C
DO 3 K=1,N
V=A(I,K)
A(I,K)=A(I1,K)
A(I1,K)=V
3 CONTINUE
4 CONTINUE
5 CONTINUE
RETURN
C
C .....
C
END

```

```

C
C *****
C * ESTA SUBROUTINA MODIFICA EL ORDEN DE DESCRIPCION DE LA ENVOLVEN-
C * TE CUANDO EL NUEVO POLIGONO A FUNDIR PUSEE TODOS SUS PUNTOS Y LA
C * FRONTERA EN EL POLIGONO NUEVO, TAL COMO SE DEFINE DE MANERA STAN-
C * DARD, NO ES COHERENTE.
C *****
C
C SUBROUTINE JTHWAY(N,SIOTH,IPOL)
C
C COMMON/F/P(100,100),E(1000,2),A(50,2),DISMAX
C COMMON/G/LP(50),M(50),NP,LE,INFUR,OUT
C COMMON/H/IE(1000),IA(50)
C
C INTEGER P,E,A,DISMAX,OUT
C LOGICAL SIOTH
C
C DO 1 J=1,N-1
C IF(IA(J).EQ.1) IA(J)=LP(IPOL)+1
C IF(ABS(IA(J)-IA(J+1)).EQ.1) GOTO 1
C GOTO 2
C 1 CONTINUE
C RETURN
C
C 2 CONTINUE
C IGIR=N-1
C CALL CLUCK(IGIR,0)
C SIOTH=.TRUE.
C RETURN
C *****
C
C END

```

```

C
C *****
C * SUBROUTINA QUE DIBUJA UNA PIRAMIDE SIMPLE O SUPERPUESTA, EN FUN-
C * CION DE UNA VARIABLE LOGICA (SUPER). SI SUPER=.FALSE., PIRAMIDE
C * SIMPLE. SI SUPER=.TRUE., PIRAMIDE SUPERPUESTA.
C *****
C
C SUBROUTINE PIRAM(RL,RL1,SEPA1,X1,Y1,PM,PF,II,NU,SUPER,J)
C
C RL = DIMENSION X EN CM DEL RECTANGULO QUE ENMARCA EL DIBUJO DE LA
C PIRAMIDE.
C RL1 = DIMENSION Y EN CM DEL RECTANGULO QUE ENMARCA EL DIBUJO DE LA
C PIRAMIDE.
C SEPA1 = MARGEN DEL DIBUJO RESPECTO DEL CUADRO.
C X1 = COORDENADA X DEL VERTICE INFERIOR IZQUIERDO DEL MARCO.
C Y1 = COORDENADA Y DEL VERTICE INFERIOR IZQUIERDO DEL MARCO.
C PM = VECTOR DONDE SE ENCUENTRAN ALMACENADOS LOS EFECTIVOS MASCU-
C LINOS DE CADA GRUPO DE EDAD.
C PF = VECTOR DONDE SE ENCUENTRAN ALMACENADOS LOS EFECTIVOS FEME-
C NINOS DE CADA GRUPO DE EDAD.
C II = NUMERO DE GRUPOS DE EDAD.
C NU = AMPLITUD EN A"OS DE LOS GRUPOS DE EDAD.
C SUPER = VARIABLE LOGICA.
C J = NUMERO DE ORDEN DE LA PIRAMIDE A DIBUJAR.
C *****
C
C DIMENSION PM(II),PF(II)
C COMMON/C/PARAM(5,20)
C COMMON/D/ESCAL,SEPAR,ESCA,SEPAR1,BASE,SEPAR2
C COMMON/E/XS(4,180),YS(4,180)
C INTEGER OUT
C LOGICAL SUPER
C
C LAS POSICIONES DE MEMORIA DE LOS ARRAYS DE LA ZONA COMUN E, SON VA-
C RIABLES INTERNAS DE DIBUJO.
C *****
C
C INICIALIZACION DE LA VARIABLE DE ESCRITURA.
C
C DATA OUT/6/
C *****
C *****
C *****
C *****
C
C PREPARACION DE LA INFORMACION Y CALCULO DE LOS PARAMETROS DE DIBUJO.
C
C IF(PARAM(5,J).EQ.0)GOTO 50
C DO 51 I=1,II
C PM(I)=PM(I)/NU
C 51 PF(I)=PF(I)/NU
C 50 SEPAR=SEPA1
C IF(SUPER)GOTO1

```

```

CALL PLOT (X1,Y1,-3)
1 CONTINUE
IF (PARAM(4,J).NE.0.)SEPAR=PARAM(4,J)
IF (SUPER) GOTO 10
A3=(RL-3.*SEPAR)/2.
WRITE(6,179)A3
179 FORMAT(F7.3)
I3=A3
RESTO=((RL-3*SEPAR)/2.)-I3
SEPAR1=SEPAR+RESTO
BASE=I3
BASE2=BASE/2.
I20=BASE2
BASE2=I20
IF ((BASE/2.).E0, BASE2)GO TO 801
BASE=BASE-1.
SEPAR1=SEPAR1+1.
801 CONTINUE
ALT=(2./3.)*(2.*BASE+SEPAR)
SEPARY=(RL1-ALT)/2.
ESCA=ALT/FLOAT(I1)
WRITE(OUT,100)BASE,ALT,SEPARY,ESCA,RL,RL1,SEPAR,SEPAR1
100 FORMAT(8F10.3)
I11=I1+1
PF(I11)=0.
CALL SCALE(PF,BASE,I11,1)
PM(I11)=0.
CALL SCALE(PM,BASE,I11,1)
I13=I1+3
ESCAL=AMAX1(PF(I13),PM(I13))
IF (PARAM(2,J).NE.0.)ESCAL=PARAM(2,J)
IF (PARAM(3,J).NE.0.)ESCA=PARAM(3,J)
IF (PARAM(3,J).NE.0.)ALT=ESCA*I1
C
C*****
C
C PINTA LOS EJES HORIZONTALES Y VERTICALES.
C
CALL AXIS(SEPAR1,SEPARY,'HOMBRES',-7,BASE,0.,(BASE+ESCAL),-ESCAL)
CALL AXIS((SEPAR1+SEPAR+BASE),SEPARY,'MUJERES',-7,BASE,0.,0.,ESCAL)
*)
CALL PLOT((SEPAR1+SEPAR+BASE),SEPARY,3)
CALL PLOT((SEPAR1+SEPAR+BASE),(SEPARY+ALT),2)
CALL PLOT((SEPAR1+BASE),(SEPARY+ALT),3)
CALL PLOT((SEPAR1+BASE),SEPARY,2)
ESCAL=1./ESCAL
10 CONTINUE
CALL PLOT((SEPAR1+BASE+SEPAR/2.),SEPARY,-3)
C
C*****
C
C PINTA ESCALONES DE LA PIRAMIDE.
C
DO 5 I=1,I1
ITRMM=2*I-1
ITRAF=2*I
Y=ESCA*(I-1)
XM=-(PM(I)*ESCAL+SEPAR/2.)

```

```

XF=PF(1)*ESCAL+SEPAR/2.
CALL PLOT(-SEPAR/2.,Y,3)
CALL PLOT(XM,Y,2)
CALL DIFER(XM,Y,SUPER,1,ITRAMM)
CALL PLOT(XM,(Y+ESCA),2)
CALL DIFER(XM,(Y+ESCA),SUPER,2,ITRAMM)
CALL PLOT(-SEPAR/2.,(Y+ESCA),2)
IF(SUPER) CALL TRAMA(ITRAMM)
IF(SUPER) GOTO 11
TAM=(SEPAR-0.2)/5.
Y1=ESCA/2.-TAM/2.+ESCA*(1-1)
IF(NU.E0.1)GOTO 35
RINF=NU*(1-1)
RSUP=NU*1-1
IF(RINF.LT.10.) X1=-TAM+TAM/2.
IF(RINF.GE.10.) X1=-(2*TAM+TAM/2.)
CALL NUMBER(X1,Y1,TAM,RINF,0.,-1)
CALL SYMBOL(999.,999.,TAM,'-',0.,1)
CALL NUMBER(999.,999.,TAM,RSUP,0.,-1)
GOTO 36
35 X1=-TAM/2.
V1=FLOAT(1)
CALL NUMBER(X1,Y1,TAM,V1,0.,-1)
11 CONTINUE
36 CALL PLOT(SEPAR/2.,Y,3)
CALL PLOT(XF,Y,2)
CALL DIFER(XF,Y,SUPER,1,ITRAMF)
CALL PLOT(XF,(Y+ESCA),2)
CALL DIFER(XF,(Y+ESCA),SUPER,2,ITRAMF)
CALL PLOT(SEPAR/2.,(Y+ESCA),2)
IF(SUPER) CALL TRAMA(ITRAMF)
5 CONTINUE
C
C*****
C
C*****
C
CALL PLOT (-SEPAR1+BASE+SEPAR/2.,-SEPAR,3)
IF(PARAM(5,J).E0.0)GOTO 12
DO 52 I=1,II
PM(I)=PM(I)*NU
52 PF(I)=PF(I)*NU
12 CONTINUE
C
C*****
C
RETURN
END

```

```

C
C *****
C * SUBROUTINE DE TRAMADO INTERNO DE UNA PIRAMIDE CON VISTAS A OBTE-
C *NER PIRAMIDES COMPUESTAS SEGUN 1 O 2 CRITERIOS DE CLASIFICACION. *
C *****
C
C SUBROUTINE PIRAM2(NRY1,DESPLZ,PM,PF,II,NU,J)
C
C NRY1 = NUMERO DE RAYAS CON QUE SE TRAMA CADA ESCALON.
C DESPLZ = VARIABLE INTERNA DE DIBUJO.
C PM Y PF = VECTORES DONDE ESTAN ALMACENADOS LOS DATOS DE ESTRUCTURA
C PDR EDAD Y SEXO COHERENTES CON LA TRAMA QUE SE VA A PRO-
C DUCIR.
C II = NUMERO DE GRUPOS DE EDAD.
C NU = AMPLITUD EN A"OS DE LOS GRUPOS DE EDAD.
C J = NUMERO DE ORDEN DE LOS DATOS COHERENTES CON LA TRAMA.
C *****
C
C DIMENSION PM(II),PF(II)
C COMMON/C/PARAM(5,20)
C COMMON/D/ESCAL,SEPAR,ESCA,SEPAR1,BASE,SEPAR2
C COMMON/F/SALTO
C
C SALTO ES UNA VARIABLE INTERNA DE DIBUJO (SEPARACION ENTRE RAYAS DE
C LA TRAMA).
C *****
C
C IF(PARAM(5,J).E0.0)GOTO 50
C DO 51 I=1,II
C PM(I)=PM(I)/NU
C 51 PF(I)=PF(I)/NU
C 50 CALL PLOT((SEPAR1+BASE+SEPAR/2.),SEPAR,-3)
C *****
C
C TRAMADO DE LOS ESCALONES MASCULINOS.
C
C DO 6 I=1,II
C XM=(PM(I)*ESCAL+SEPAR/2.)
C CALL PLOT(XM,(ESCA*(I-1)),3)
C CALL PLOT(XM,(ESCA*I),2)
C XMO=SEPAR/2.
C SALTO=ESCA/FLOAT(NRY1+1)
C Y=SALTO+ESCA*(I-1)-SALTO*DESPLZ
C IF (DESPLZ.E0.0.5) GOTO 700
C GOTO 701
C 700 NRY1=NRY1+1
C 701 CONTINUE
C DO 7 IRAY=1,NRY1,2
C CALL PLOT(XMO,(Y+SALTO*(IRAY-1)),3)
C CALL PLOT(XM,(Y+SALTO*(IRAY-1)),2)
C IF (IRAY.E0.NRY1) GOTO 7
C CALL PLOT(XM,(Y+SALTO*IRAY),3)
C CALL PLOT(XMO,(Y+SALTO*IRAY),2)
C 7 CONTINUE
C IF(DESPLZ.E0.0.5) NRY1=NRY1-1

```

```

6 CONTINUE
C
C*****
C
C TRAMADO DE LOS ESCALONES FEMENINOS.
C
DO 8 I=1,11
XF=PF(I)*ESCAL+SEPAR/2.
CALL PLOT (XF,(ESCA=(I-1)),3)
CALL PLOT (XF,(ESCA=I),2)
XFD=SEPAR/2.
Y=SALTO+ESCA*(I-1)-SALTO*DESPLZ
IF(DESPLZ.EQ.0.5) GOTO 600
GOTO 601
600 NRY1=NRY1+1
601 CONTINUE
DO 9 IRAY=1,NRY1,2
CALL PLOT(XFD,(Y+SALTO*(IRAY-1)),3)
CALL PLOT(XF,(Y+SALTO*(IRAY-1)),2)
IF (IRAY.EQ.NRY1) GOTO 9
CALL PLOT(XF,(Y+SALTO*IRAY),3)
CALL PLOT(XFD,(Y+SALTO*IRAY),2)
9 CONTINUE
IF(DESPLZ.EQ.0.5) NRY1=NRY1-1
8 CONTINUE
C
C*****
C
IF(PARAM(5,J).EQ.0)GOTO 60
DO 61 I=1,11
PM(I)=PM(I)*NU
61 PF(I)=PF(I)*NU
60 CONTINUE
CALL PLOT(-(SEPAR1+BASE+SEPAR/2.),-SEPAR,-3)
C
C*****
C
RETURN
END

```



```

C
C *****
C *   ESTA SUBROUTINA DIBUJA LINEAS AL ESTILO DE *PLOT* QUE UNEN LAS *
C *PROYECCIONES DE PUNTOS EN EL ESPACIO. *
C *   SIEMPRE SE PROYECTA SOBRE EL PLANO Y-Z. *
C *****
C
C   SUBROUTINE PLOT3(X,Y,Z,IP)
C
C   X = COORDENADA X EN EL ESPACIO ORIGEN.
C   Y = COORDENADA Y EN EL ESPACIO ORIGEN.
C   Z = COORDENADA Z EN EL ESPACIO ORIGEN.
C   IP = VARIABLE DE CONTROL DE LA PLUMA.
C *****
C
C   COMMON/A/A(3,3)
C
C   A = MATRIZ DE TRANSFORMACION CALCULADA EN VISTA.
C *****
C
C   XP=X*A(1,1)+Y*A(2,1)+Z*A(3,1)
C   YP=X*A(1,2)+Y*A(2,2)+Z*A(3,2)
C   ZP=X*A(1,3)+Y*A(2,3)+Z*A(3,3)
C   XS=YP
C   YS=ZP
C   CALL PLDTMM(XS,YS,IP)
C   RETURN
C *****
C
C   END

```

```

C
C *****
C *   ESTA SUBROUTINA DIBUJA AL ESTILO DE PLOT, PERO CONSERVANDO EN LA
C *ZONA COMUN XYMM LOS VALORES DE LOS VERTICES DEL MINIMO RECTANGULO *
C *QUE ENMARCA EL DIBUJO.
C *****
C
C   SUBROUTINE PLOTMM(X,Y,IP)
C
C   X = COORDENADA X DEL PUNTO AL QUE SE MUEVE LA PLUMA.
C   Y = COORDENADA Y DEL PUNTO AL QUE SE MUÈVE LA PLUMA.
C   IP = VARIABLE DE CONTROL DE LA PLUMA.
C *****
C
C   COMMON/XYMM/XMAX,YMAX,XMIN,YMIN,FCT
C
C   ANTES DE EMPEZAR CADA DIBUJO HAY QUE INICIALIZAR XMAX E YMAX
C   CON UN VALOR MINIMO (-1.E74) Y XMIN E YMIN CON UN VALOR MAXI
C   MO (1.E74).
C *****
C
C   TEST PARA CONOCER EL RECUADRO.
C
C   XF=X*FCT
C   YF=Y*FCT
C
C   IF(XF.GT.XMAX)XMAX=XF
C   IF(XF.LT.XMIN)XMIN=XF
C   IF(YF.GT.YMAX)YMAX=YF
C   IF(YF.LT.YMIN)YMIN=YF
C
C   DIBUJA RECURRIENDO A PLOT.
C
C   CALL PLOT(X,Y,IP)
C   RETURN
C *****
C
C   END

```

```

C
C *****
C *   ESTA SUBROUTINA DIBUJA AL ESTILO DE PLOT, PERO CONSERVANDO EN LA *
C * ZONA COMUN XYMM LOS VALORES DE LOS VERTICES DEL MINIMO RECTANGULO *
C * QUE ENMARCA EL DIBUJO (SIN CONSIDERAR EL FACTOR DE ESCALA DEL DI- *
C * BUJO). *
C *****
C
C   SUBROUTINE PLOTM2(X,Y,IP)
C
C   X = COORDENADA X DEL PUNTO AL QUE SE MUEVE LA PLUMA.
C   Y = COORDENADA Y DEL PUNTO AL QUE SE MUEVE LA PLUMA.
C   IP = VARIABLE DE CONTROL DE LA PLUMA.
C *****
C
C   COMMON/XYMM/XMAX,YMAX,XMIN,YMIN
C
C   ANTES DE EMPEZAR CADA DIBUJO HAY QUE INICIALIZAR XMAX E YMAX
C   CON UN VALOR MINIMO (-1.E74) Y XMIN E YMIN CON UN VALOR MAXI
C   MO (1.E74).
C *****
C
C   TEST PARA CONOCER EL RECUADRO.
C
C   IF(X.GT.XMAX)XMAX=X
C   IF(X.LT.XMIN)XMIN=X
C   IF(Y.GT.YMAX)YMAX=Y
C   IF(Y.LT.YMIN)YMIN=Y
C
C   DIBUJA RECURRIENDO A PLOT.
C
C   CALL PLOT(X,Y,IP)
C   RETURN
C *****
C
C   END

```

```

C
C *****
C *   ESTA SUBROUTINA RECONSTRUYE EL AREA J A PARTIR DE LAS LI-
C *   NEAS DIGITIZADAS ALMACENADAS EN COORD Y DE LOS CODIGOS DOBLES ASD-
C *   CIADOS A CADA LINEA (AREA A LA DERECHA, AREA A LA IZQUIERDA) ALMA-
C *   CENADOS EN ICOD.
C *****
C
C   SUBROUTINE POLIG(J,LPOLY)
C
C   LPOLY = NUMERO DE PUNTOS QUE DESCRIBEN EL AREA J.
C *****
C
COMMON/A/COORD(200,100),POLY(500,2)
COMMON/B/LAR(100),ICOD(2,100),NAR,NLIN
INTEGER IPUNT(20)
DATA INFOR/0/
K=0
DO 1 I=1,NLIN
IF(ICOD(1,I).EQ.J)
*OR.ICOD(2,I).EQ.J)GOTO 2
GOTO 1
2 K=K+1
IPUNT(K)=I
IF(ICOD(2,I).NE.J)GOTO 1
CALL INVERT(COORD(1,I),LAR(I))
IC=ICOD(1,I)
ICOD(1,I)=ICOD(2,I)
ICOD(2,I)=IC
1 CONTINUE
K1=0
J1=IPUNT(1)
KLO=0
KL=0
3 CONTINUE
K1=K1+1
KLO=KLO+KL
L=LAR(J1)
IA=0
DO 4 I=1,L,2
IA=IA+1
KLIA=KLO+IA
IM1=I+1
IF(INFOR.NE.0)
*WRITE(6,200) KLIA,I,IM1,J1
200 FORMAT(4I6)
POLY(KLIA,1)=COORD(I,J1)
POLY(KLIA,2)=COORD(IM1,J1)
4 CONTINUE
IF(INFOR.NE.0)
*WRITE(6,201)
201 FORMAT(' HA SALIDO POR 4 CONTINUE')
KL=IA
DO 5 I1=1,K
IF(ABS(COORD(L,J1)-COORD(2,IPUNT(I1))).LT.0.0001)
*AND,ABS(COORD(L-1,J1)-COORD(1,IPUNT(I1))).LT.0.0001)GOTO 6
5 CONTINUE

```

```
6 J1=IPUNT(I)
  IF(K1.EQ.K) GOTO 7
  GOTO 3
7 CONTINUE
  LPOLY=KLO+KL
  CALL DEPURA(LPOLY)
  RETURN
C .....
C
END
```

```

C
C *****
C *   ESTA SUBROUTINA RECONSTRUYE Y ALMACENA EN POLY LA DESCRIPCION *
C *   POR COORDENADAS DE LOS PUNTOS QUE DEFINEN LA ZONA RESULTADO DE LA *
C *   FUSION DEL AREA J Y DEL AREA J2, A PARTIR DE LAS LINEAS DE FRON- *
C *   TERA ENTRE TODAS LAS AREAS EN CUESTION Y DE LOS CODIGOS ASOCIADOS *
C *   A ESTAS. *
C *****
C
C   SUBROUTINE POLIG2(J,J2,LPOLY)
C
C   LPOLY = NUMERO DE PUNTOS QUE DESCRIBEN EL AREA QUE RESULTA.
C *****
C
C   COMMON/A/COORD(200,100),POLY(500,2)
C   COMMON/B/LAR(100),ICOD(2,100),NAR,NLIN
C   INTEGER IPUNT(20)
C   DATA INFOR/0/
C   K=0
C   DO 1 J=1,NLIN
C     IF((ICOD(1,J).EQ.J.AND.ICOD(2,J).EQ.J2).
C *OR.(ICOD(1,J).EQ.J2.AND.ICOD(2,J).EQ.J))GOTO 1
C     IF((ICOD(1,J).EQ.J.OR.ICOD(1,J).EQ.J2).
C *OR.(ICOD(2,J).EQ.J.OR.ICOD(2,J).EQ.J2))GOTO 2
C     GOTO 1
C   2 CONTINUE
C
C   SOLO SE CONSIDERAN LAS LINEAS QUE SON FRONTERA ENTRE J1 Y OTRA AREA
C   DISTINTA DE J2 Y LAS QUE SON FRONTERA ENTRE J2 Y OTRA AREA DISTIN-
C   TA DE J1.
C
C   K=K+1
C   IPUNT(K)=J
C   IF(ICOD(2,J).NE.J.AND.ICOD(2,J).NE.J2)GOTO 1
C   CALL INVERT(COORD(1,J),LAR(J))
C   IC=ICOD(1,J)
C   ICD(1,J)=ICOD(2,J)
C   ICD(2,J)=IC
C 1 CONTINUE
C   K1=0
C   J1=IPUNT(1)
C   KLO=0
C   KL=0
C 3 CONTINUE
C   K1=K1+1
C   KLO=KLO+KL
C   L=LAR(J1)
C   IA=0
C   DO 4 I=1,L,2
C     IA=IA+1
C     KLIA=KLO+IA
C     IMI=I+1
C     IF(INFOR.NE.0)
C *WRITE(6,200) KLIA,I,IMI,J1
C 200 FORMAT(4I6)
C     POLY(KLIA,1)=COORD(I,J1)
C     POLY(KLIA,2)=COORD(IMI,J1)

```

```
4 CONTINUE
  IF(INFOR.NE.0)
    *WRITE(6,201)
201 FORMAT(' HA SALIDO POR 4 CONTINUE')
  KL=IA
  DO 5 I1=1,K
    IF(ABS(COORD(L,J1)-COORD(2,I1PUNT(I1)))>.0001.
    *AND(ABS(COORD(L-1,J1)-COORD(1,I1PUNT(I1)))>.0001)GOTO 6
  5 CONTINUE
  6 J1=I1PUNT(I1)
    IF(K1.EQ.K) GOTO 7
    GOTO 3
  7 CONTINUE
  LPOLY=KLO+KL
  CALL DEPURA(LPOLY)
  RETURN
C
C*****
C
  END
```

```

C
C *****
C *   ESTA SUBROUTINA RECONSTRUYE EL AREA CUYO CODIGO SE ENCUENTRA AL-
C *MACENADO EN J(5) A PARTIR DE LAS LINEAS DE FRONTERA ENTRE LAS UNI-
C *DADES ESPACIALES MAS DESAGREGADAS. *
C *****
C
C   SUBROUTINE POLIG3
C   COMMON/A/COORD(200,100),POLY(500,2)
C   COMMON/B/LAR(100),ICOD(2,100,5),J(5),NAR1,NAR2,NAR3,NAR4,
C   *NAR5,NLIN,ISW,LPOLY
C   INTEGER IPUNT(20)
C   LOGICAL DER,IZ0
C   DATA INFOR/0/
C
C SELECCION DE LAS LINEAS QUE FORMAN EL POLIGONO CUYA CLAVE SE EN-
C CUENTRA ALMACENADA EN J.
C ISW=1, CONTORNOS DE REGIONES AUTONOMICAS.
C ISW=2, CONTORNOS DE PROVINCIAS.
C ISW=3, CONTORNOS DE MUNICIPIOS.
C ISW=4, CONTORNOS DE DISTRITOS MUNICIPALES.
C ISW=5, CONTORNOS DE SECCIONES CENSALES.
C
C   K=0
C   DO 1 I=1,NLIN
C   CALL TEST(I,DER,IZ0)
C   IF(DER) GOTO 2
C   IF(IZ0) GOTO 2
C   GOTO 1
C
C SE CONSIDERA LA LINEA.
C
C   2 K=K+1
C   IPUNT(K)=I
C   IF(DER)GOTO 1
C
C SE INVIERTE LA LINEA, QUE ESTABA A IZQUIERDAS, Y SE REORDENAN COHE-
C RENTEMENTE LOS CODIGOS.
C
C   CALL INVERT(COORD(1,I),LAR(I))
C   DO 8 KI=1,ISW
C   IC=ICOD(1,I,KI)
C   ICD(1,I,KI)=ICOD(2,I,KI)
C   ICD(2,I,KI)=IC
C 8 CONTINUE
C   IF(INFOR.NE.0)
C   *WRITE(6,206)(COORD(JKL,I),JKL=1,LAR(I))
C 206 FORMAT(' COORD'/(1X,16F6.2))
C 1 CONTINUE
C   IF(INFOR.NE.0)
C   *WRITE(6,205)(IPUNT(I),I=1,K)
C 205 FORMAT(' IPUNT'/(1X,30I4))
C   K2=0
C   J1=IPUNT(1)
C   KL=0
C   KL=0
C 3 CONTINUE
C   IF(INFOR.NE.0)

```



```

*WRITE(6,203) J1
203 FORMAT(' J1=',I3)
K2=K2+1
KLO=KLO+KL
L=LAR(J1)
IA=0
IF(INFOR.NE.0)
*WRITE(6,202) L
202 FORMAT(' L=',I3)
DO 4 I=1,L,2
IA=IA+1
KLIA=KLO+IA
IM1=I+1
IF(INFOR.NE.0)
*WRITE(6,200) KLIA,I,IM1,J1
200 FORMAT(4I6)
POLY(KLIA,1)=COORD(I,J1)
POLY(KLIA,2)=COORD(IM1,J1)
4 CONTINUE
IF(INFOR.NE.0)
*WRITE(6,201)
201 FORMAT(' HA SALIDO POR 4 CONTINUE')
KL=IA
DO 5 I1=1,K
IF(ABS(COORD(L,J1)-COORD(2,IPUNT(I1)))>.LT.0.001)
*AND,ABS(COORD(L-1,J1)-COORD(1,IPUNT(I1)))>.LT.0.001)GOTO 6
5 CONTINUE
IF(INFOR.NE.0)
*WRITE(6,204) I1
204 FORMAT(' I1=',I3)
6 J1=IPUNT(I1)
IF(K2.E0.K) GOTO 7
GOTO 3
7 CONTINUE
LPOLY=KLO+KL
CALL DEPURA(LPOLY)
RETURN
C
C*****
C
END

```

```

C
C *****
C * SUBROUTINA DE CALCULO DE PORCENTAJES DE LOS GRUPOS DE EDAD DE *
C * UNA POBLACION RESPECTO DE SU PROPIO TOTAL. *
C *****
C
C SUBROUTINE PORCEN(PM,PF,N,DATMA,DATFA)
C
C PM Y PF = VECTORES EN LOS QUE ESTAN ALMACENADOS LOS DATOS DE ES-
C TRUCTURA POR EDAD Y SEXO.
C N = NUMERO DE GRUPOS DE EDAD.
C DATMA = VECTOR AUXILIAR.
C DATFA = VECTOR AUXILIAR.
C *****
C
C DIMENSION PM(N),PF(N),DATMA(N),DATFA(N)
C *****
C
C DO 3 I=1,N
C DATMA(I)=PM(I)
C DATFA(I)=PF(I)
C 3 CONTINUE
C POBTOT=0.
C DO 1 I=1,N
C 1 POBTOT=POBTOT+PM(I)+PF(I)
C DO 2 I=1,N
C PM(I)=PM(I)/POBTOT*100.
C 2 PF(I)=PF(I)/POBTOT*100.
C *****
C
C RETURN
C END

```

```

C
C
C *****
C * SUBROUTINA DE CALCULO DE PORCENTAJES DE LOS GRUPOS DE EDAD DE *
C * UNA POBLACION RESPECTO DEL TOTAL DE OTRA POBLACION QUE LA COMPREN- *
C * DE. *
C *****
C
C SUBROUTINE PORCE2(PM,PF,N,DATMA,DATFA)
C
C PM Y PF = VECTORES EN LOS QUE ESTAN ALMACENADOS LOS DATOS DE ES-
C TRUCTURA POR EDAD Y SEXO.
C N = NUMERO DE GRUPOS DE EDAD.
C DATMA = VECTOR DONDE ESTAN ALMACENADOS LOS EFECTIVOS MASCULINOS
C DE LA POBLACION DE REFERENCIA.
C DATFA = VECTOR DONDE ESTAN ALMACENADOS LOS EFECTIVOS FEMENINOS
C DE LA POBLACION DE REFERENCIA.
C *****
C DIMENSION PM(N),PF(N),DATMA(N),DATFA(N)
C *****
C
C POBTOT=0.
C DO 1 I=1,N
1 POBTOT=POBTOT+DATMA(I)+DATFA(I)
C DO 2 I=1,N
C PM(I)=PM(I)/POBTOT*100.
2 PF(I)=PF(I)/POBTOT*100.
C *****
C
C RETURN
C END

```

```

C
C *****
C * ESTA SUBROUTINA AVERIGUA LOS PUNTOS DE CORTE DE UN POLIGONO, *
C * DEFINIDO POR LAS COORDENADAS DE SUS VERTICES ALMACENADAS EN LOS *
C * VECTORES X E Y , CON LA RECTA Y=YR -PARALELA AL EJE X-, Y LOS DE- *
C * VUELVE ORDENADOS, EN FUNCION DE SU COORDENADA X, EN EL VECTOR A. *
C *****
C
C SUBROUTINE RAY(X,Y,N,YR,A,K)
C DIMENSION A(1),X(N),Y(N)
C *****
C SE CALCULAN LOS PUNTOS DE CORTE DE LA RAYA CON EL POLIGONO.
C
C K=0
C DO 7 I=1,N
C X1=X(I)
C Y1=Y(I)
C I1=I+1
C IF(I1.GT.N) I1=1
C X2=X(I1)
C Y2=Y(I1)
C IF(YR.LE.Y2.AND.YR.GT.Y1) GO TO 5
C IF(YR.GE.Y1.OR.YR.LT.Y2) GO TO 6
C 5 CONTINUE
C XC=X1+(YR-Y1)*(X2-X1)/(Y2-Y1)
C K=K+1
C A(K)=XC
C 6 CONTINUE
C 7 CONTINUE
C *****
C IF(K.EQ.0) GO TO 20
C *****
C SE ORDENAN LOS PUNTOS DE CORTE EN FUNCION DE LA COORDENADA X.
C
C K1=K-1
C DO 11 I=1,K1
C IMIN=I
C I1=I+1
C DO 10 J=I1,K
C IF(A(IMIN).GT.A(I1)) IMIN=I1
C 10 CONTINUE
C V=A(IMIN)
C A(IMIN)=A(I)
C A(I)=V
C 11 CONTINUE
C *****
C 20 CONTINUE
C *****
C

```

362

RETURN  
END

```

C
C *****
C * ESTA SUBROUTINA AVERIGUA LOS PUNTOS DE CORTE DE UN POLIGONO, DE-
C *FINIDO POR LAS COORDENADAS DE SUS VERTICES ALMACENADAS EN LOS VEC-
C *TORES X E Y, CON LA RECTA Y=YR -PARALELA AL EJE X-, Y LOS DEVUELVE*
C *ORDENADOS, EN FUNCION DE SU COORDENADA X, EN EL VECTOR A. *
C *****
C
C SUBROUTINE RAY(X,Y,N,YR,A,K)
C DIMENSION A(1),X(N),Y(N)
C DATA INFOR/1/
C
C *****
C *****
C *****
C SE CALCULAN LOS PUNTOS DE CORTE DE LA RAYA CON EL POLIGONO.
C
C 15 CONTINUE
C K=0
C IK=0
C IKMEM=0
C IMEM=0
C IIMEM=0
C *****
C DO 7 I=1,N
C *****
C X1=X(I)
C Y1=Y(I)
C II=I+1
C IF(II.GT.N) II=1
C X2=X(II)
C Y2=Y(II)
C IF(YR.LE.Y2.AND.YR.GE.Y1) GO TO 5
C IF(YR.GT.Y1.OR.YR.LT.Y2) GO TO 7
C *****
C 5 CONTINUE
C *****
C 5.1. LA RAYA CORTA AL SEGMENTO QUE SE ESTA CONSIDERANDO.
C
C IF(INFOR.NE.1)
C *WRITE(6,100) I
C 100 FORMAT(I3)
C
C 5.2. SE CALCULA LA COORDENADA X DEL PUNTO DE CORTE.
C
C IF(Y2.EQ.Y1)GOTO 22
C XC=X1+(YR-Y1)*(X2-X1)/(Y2-Y1)
C GOTO 21
C 22 XC=X1
C 21 CONTINUE
C
C IF(I.EQ.1) GOTO 16
C
C 23 CONTINUE
C IF(INFOR.NE.1)
C *WRITE(6,997)IK,IKMEM
C 997 FORMAT(2I4)

```

```

C
  IF(I.NE.IIMEM)          GOTD 8
  IF(IK.EQ.0.AND.IKMEM.NE.0) GOTD 12
  IF((Y2-Y1)*(Y(IIMEM)-Y1)) 7,9,8
C
C*****
  8 CONTINUE
C*****
C 8. SE CONSIDERA EL PUNTO DE CORTE.
C
  K=K+1
  IMEM=I
  IIMEM=I
  A(K)=XC
  GOTD 7
C*****
  16 CONTINUE
C*****
C 16. CUANDO LA RAYA CORTA EL PRIMER SEGMENTO, SE INVESTIGA SI AQUELLA
C LO HACE A LA ALTURA DEL COMIENZO DE DICHO SEGMENTO
C
  IF(YR.NE.Y1)          GOTD 23
  IF((Y2-Y1)*(Y(N)-Y1)) 7,9,8
C
C*****
  9 CONTINUE
C*****
C 9. EL SEGMENTO ES COLINEAL A LA RAYA.
C
  IF(I.EQ.1)          GOTD 13
C
  IF(IK.EQ.0) IPUNT=I
  IK=IK+1
  A(K+1)=XC
  IMEM=I
  IIMEM=I
  IF(INFOR.NE.1)
  *WRITE(6,995) Y(II+1),Y(II)
  995 FORMAT(2F7.2)
C
  IF(Y(II+1).EQ.Y(II)) GOTD 7
C
C*****
  6 CONTINUE
C*****
C 6. YA NO HAY MAS SEGMENTOS COLINEALES.
C
  IKMEM=IK
  IK=0
  GOTD 7
C*****
  12 CONTINUE
C*****
C 12. TRATAMIENTO DEL SIGUIENTE SEGMENTO A LOS COLINEALES.
C
  IKMEM=0
  YREF1=Y(IPUNT-1)
  YREF2=Y(IPUNT)

```

```

      IF((YREF1-YREF2)*(Y2-Y1).GT.0.) GOTO 8
C*****
      7 CONTINUE
C*****
      GOTO 14
C*****
      13 CONTINUE
C*****
C 13. REORGANIZACION DE LAS COORDENADAS DEL POLIGONO SI HAY PROBLEMAS
C DE COLINEALIDAD EN EL PRIMER O EN EL ULTIMO SEGMENTO
C
      CALL REORG(X,Y,N)
      IF(INFOR.NE.1)
      *WRITE(6,999)(X(I32),Y(I32),I32=1,N)
      999 FORMAT(1X,2F6.2)
      GOTO 15
C
C*****
C*****
C*****
C
      14 CONTINUE
      IF(K.EQ.0) GO TO 20
C*****
C
C SE ORDENAN LOS PUNTOS DE CORTE EN FUNCION DE LA COORDENADA X.
C
      K1=K-1
      DO 11 I=1,K1
      IMIN=I
      I1=I+1
      DO 10 II=I1,K
      IF(A(IMIN).GT.A(II)) IMIN=II
      10 CONTINUE
      V=A(IMIN)
      A(IMIN)=A(I)
      A(I)=V
      11 CONTINUE
C
C*****
C
      20 CONTINUE
      RETURN
C*****
C
      END

```



```

C
C *****
C * ESTE SUBROUTINA INCLUYE ADECUADAMENTE, DENTRO DE LA ENVOLVENTE *
C * ANTERIOR, LA SECCION DEL POLIGONO QUE SE HA DE FUNDIR A ELLA PARA *
C * CREAR LA NUEVA ENVOLVENTE. *
C *****
C
C SUBROUTINE REINSC(13,14,1)
C
C COMMON/F/P(100,100),E(1000,2),A(50,2),DISMAX
C COMMON/G/LP(50),M(50),NP,LE,INFUR,OUT
C
C INTEGER OUT,P,E,A,DISMAX
C
C 15=14-13-1
C 16=1-15
C
C IF(16) 1,3,5
C
C PARA REDUCIR EL HUECO.
C
C 1 CONTINUE
C K=14
C 2 CONTINUE
C K2=K+16
C E(K2,1)=E(K,1)
C E(K2,2)=E(K,2)
C IF(K.EQ.LE) GOTO 3
C K=K+1
C GOTO 2
C
C PASAR DE A A E.
C
C 3 CONTINUE
C DO 4 J=1,1
C 13MJ=13+J
C E(13MJ,1)=A(J,1)
C E(13MJ,2)=A(J,2)
C 4 CONTINUE
C LE=LE+16
C RETURN
C *****
C
C PARA HACER HUECO.
C
C 5 CONTINUE
C L=LE+16
C 6 CONTINUE
C LMI=L-16
C E(L,1)=E(LMI,1)
C E(L,2)=E(LMI,2)
C IF(LMI.EQ.14) GOTO 3
C L=L-1
C GOTO 6
C
C *****
C
C END

```

```

C
C *****
C * ESTA SUBROUTINA SOLUCIONA LOS PROBLEMAS DE COLINEALIDAD -CON UNA*
C *RAYA QUE CORTE A UN POLIGONO, DESCRITO POR LAS COORDENADAS DE SUS *
C *VERTICES- EN EL PRIMER O EN EL ULTIMO SEGMENTO, MODIFICANDO EL OR-*
C *DEN DE LA LISTA DE LOS VERTICES SIN ALTERAR LA DESCRIPCION DEL *
C *POLIGONO. *
C *****
C
SUBROUTINE REORG(X,Y,N)
DIMENSION X(N),Y(N),RMEM(2)
DO 1 I=1,N
  JJ=I+1
  IM1=I-1
  IF(I.EQ.1)IM1=N
  IF((Y(IJ)-Y(I))*(Y(I)-Y(IM1)).NE.0.) GOTO 2
1 CONTINUE
2 CONTINUE
DO 3 J=1,I-1
  RMEM(1)=X(J)
  RMEM(2)=Y(J)
DO 4 L=N-1,N-1
  LM1=L+1
  X(L)=X(LM1)
  Y(L)=Y(LM1)
4 CONTINUE
X(N)=RMEM(1)
Y(N)=RMEM(2)
3 CONTINUE
RETURN
C
C *****
C
END

```

```

C
C *****
C *   ESTA SUBROUTINA SITUA UN ROTULO EXPLICATIVO DE 77 CARACTERES, *
C *   COMO MAXIMO, EN LA BASE DE LA PIRAMIDE. *
C *****
C
SUBROUTINE ROTULO(RLX,J,ROT)
COMMON/B/ROTUL(80,21)
COMMON/D/ESCAL,SEPAR,ESCA,SEPAR1,BASE,RLY
CHARACTER ROTUL
LOGICAL ROT
ROT=.TRUE.
TAM=0.5
IF((TAM*77.).GT.(RLX-4.)) GOTO 1
GOTO 2
1 CONTINUE
TAM=(RLX-4.)/77.
2 CONTINUE
RLY2=RLY/2.
RMARG=(RLX-TAM*77.)/2.
W=0.
NC=77
CALL SYMBOL(RMARG,(RLY2-TAM/2.),TAM,ROTUL(1,J),W,NC)
C *****
C
RETURN
END

```

```

C
C *****
C * ESTA SUBROUTINA SITUA UNA ESCALA VERTICAL PARA LECTURA DE LOS *
C * BLOQUES DIAGRAMA, SOBRE EL PUNTO (X,Y). *
C * EN LA VERSION ACTUAL DIVIDE EL RECORRIDO EN 5 PARTES. *
C *****
C
C SUBROUTINE SCVERT(X,Y,ZB,Z0,Z1,ZF,ZINC,XTOP,YTOP)
C
C ZB = VALOR DE LA FUNCION FUERA DEL CONTORNO.
C Z0 = VALOR MINIMO DE LA FUNCION DENTRO DEL CONTORNO.
C Z1 = VALOR MAXIMO DE LA FUNCION DENTRO DEL CONTORNO.
C ZF = FACTOR DE CORRECCION DE LA FUNCION (*).
C ZINC = FACTOR DE CORRECCION DE LA FUNCION (+).
C XTOP = COORDENADA X MAXIMA QUE RECORRE LA PLUMA EN ESTE DIBUJO.
C YTOP = COORDENADA Y MAXIMA QUE RECORRE LA PLUMA EN ESTE DIBUJO.
C *****
C
C INICIALIZACION DE VARIABLES.
C
C DATA NINT/5/
C *****
C
C SE SITUAN EN LA ESCALA LOS VALORES ZB,Z0 Y Z1
C
C CALL PLOT(X,Y,-3)
C CALL PLOT3(0.,0.,0.,3)
C CALL PLOT3(0.,0.,ZB,2)
C CALL WHERE(X0,Y0,RFACT)
C CALL PLOT(X0+0.5,Y0,2)
C CALL SYMBOL(X0-1.5,Y0,0.5,'ZB',0.,2)
C CALL PLOT(X0,Y0,3)
C CALL PLOT3(0.,0.,Z0,2)
C CALL WHERE(X1,Y1,RFACT)
C CALL PLOT(X1+0.5,Y1,2)
C VALUE1=Z0/ZF-ZINC
C CALL NUMBER(X1+1.,Y1,0.5,VALUE1,0.,2)
C CALL SYMBOL(999.,999.,0.5,'Y MENOS',0.,8)
C CALL WHERE(XMAX1,Y10,RFACT)
C CALL PLOT(X1,Y1,3)
C CALL PLOT3(0.,0.,Z1,2)
C CALL WHERE(X2,Y2,RFACT)
C CALL PLOT(X2+0.5,Y2,2)
C VALUE2=Z1/ZF-ZINC
C CALL NUMBER(X2+1.,Y2,0.5,VALUE2,0.,2)
C CALL SYMBOL(999.,999.,0.5,'Y MAS',0.,6)
C YMAX=Y2+0.5
C CALL WHERE(XMAX2,Y10,RFACT)
C XMAX=AMAX1(XMAX1,XMAX2)
C *****
C
C SE SITUAN LOS VALORES INTERMEDIOS.
C
C VINC=(VALUE2-VALUE1)/NINT
C YINC=(Y2-Y1)/NINT

```

```

YDD =Y1
VDD =VALUE1
NINTM1=NINT-1
CALL PLOT(X1,Y1,3)
DO 1 I=1,NINTM1
YDD=YDD+YINC
VDD=VDD+VINC
CALL PLOT(X1,YDD,3)
CALL PLOT(X1+0.5,YDD,2)
CALL NUMBER(X1+1.,YDD,0.5,VDD,0.,2)
1 CONTINUE
C
C.....
XTOP=X+XMAX
YTOP=Y+YMAX
C.....
C
C SE DEVUELVE EL ORIGEN DE COORDENADAS A DONDE SE ENCONTRABA, ANTES
C DE LLAMAR A ESTA SUBROUTINA.
C
CALL PLOT(-X,-Y,-3)
RETURN
C
C.....
C
END

```

```
C
C *****
C *   ESTA SUBROUTINA CALCULA EL ANGULO DE RAYADO COHERENTE CON EL PUN*
C *TO DE VISTA ELEGIDO Y CON LAS PETICIONES DEL USUARIO. *
C *****
C
C   SUBROUTINE SET62(X,Y,W)
C   A=ATAN2(Y,X)
C   A=A*180./3.141593
C   W=90.+ A
C   RETURN
C *****
C
C   END
```

```

C
C *****
C * ESTA SUBROUTINA SITUA EN D EL VALOR .TRUE., SI LA LINEA I ES *
C *FRONTERA -DEJANDOLA A LA DERECHA- DEL AREA DESCRITA POR EL CODIGO *
C *ALMACENADO EN J(5) Y OTRA AREA DEL MISMO NIVEL DE AGREGACION, SI *
C *TUA EN IZ EL VALOR .TRUE, CUANDO SE PRODUCE LA MISMA SITUACION, *
C *DEJANDO EL AREA A SU IZQUIERDA, EN AMBAS VARIABLES SE ALMACENA *
C *.FALSE, CUANDO LA LINEA NO DEBE CONSIDERARSE EN LA RECONSTRUCCION *
C *DEL AREA J(5). *
C *****
C
SUBROUTINE TEST(I,D,IZ)
COMMON/B/LAR(100),ICOD(2,100,5),J(5),NAR1,NAR2,NAR3,NAR4,
+NAR5,NLIN,ISW,LPOLY
LOGICAL D,IZ
K1=0
D=.TRUE.
IZ=.TRUE.
1 CONTINUE
K1=K1+1
IF(IZ) GOTO 2
GOTO 3
2 CONTINUE
IF(ICOD(2,I,K1).NE.J(K1)) IZ=.FALSE.
3 CONTINUE
IF(D) GOTO 4
GOTO 5
4 CONTINUE
IF(ICOD(1,I,K1).NE.J(K1)) D=.FALSE.
5 CONTINUE
IF(K1.LT.ISW) GOTO 1
IF(ICOD(1,I,K1).EQ.ICOD(2,I,K1)) GOTO 6
RETURN
6 CONTINUE
IZ=.FALSE.
D=.FALSE.
RETURN
C
C *****
C
END

```

```

C
C *****
C * ESTA SUBROUTINA SITUA EN LA ZONA COMUN L EL VALOR .TRUE. CUANDO *
C *LA FRONTERA COMUN EN EL POLIGONO 'PASA POR O'. *
C *****
C
C SUBROUTINE TESTFA(N,IP)
C
C COMMON/76/LP(50),M(50),NP,LE,INFOR,OUT
C COMMON/77/IE(1500),IA(50)
C COMMON/78/RESTA
C DIMENSION ICCOC(50)
C
C LOGICAL RESTO
C INTEGER OUT
C
C IF(N.EQ.2) GOTO 6
C
C DO 1 I=1,N-1
C IF(IA(I+1)-IA(I)) 2,3,4
C 2 ICCOC(I)=1
C GOTO 1
C 3 WRITE(OUT,100) IP
C CALL EXIT
C 4 ICCOC(I)=0
C 1 CONTINUE
C DO 5 I=1,N-2
C IF(ICCOC(I+1).EQ.ICCOC(I)) GOTO 5
C RESTO = .TRUE.
C RETURN
C 5 CONTINUE
C RETURN
C
C 6 CONTINUE
C IF((IA(1).EQ.1.AND. IA(2).EQ.LP(IP))
C .OR.(IA(1).EQ.LP(IP).AND. IA(2).EQ.1)) GOTO 7
C RETURN
C 7 RESTO = .TRUE.
C RETURN
C
C *****
C
C 100 FORMAT(' HAY PROBLEMAS EN TESTFA EN EL POLIGONO N = ',I4)
C END

```



```

C
C *****
C * ESTA SUBROUTINA INVESTIGA SI LA FRONTERA EN LA ENVOLVENTE 'PASA *
C * POR O', SI ESTO ES ASI, CAMBIA EL ORDEN DE LOS PUNTOS QUE DESCRIBEN *
C * EN LA ENVOLVENTE, RECURRIENDO A LA SUBROUTINA CLOCK. *
C * ESTA SUBROUTINA TAMBIEN COMPRUEBA QUE LA FRONTERA ENVOLVENTE-POLIGONO *
C * ES 'CONTIGUA', SI LA FRONTERA NO ES DE ESTE TIPO, DEVUELVE *
C * UTRD EN .TRUE. Y CONTIG BUSCA OTRO POLIGONO PARA FUNDIR. *
C *****
C
C SUBROUTINE TESTFE(N,OTRO,SICLOC)
C
C COMMON/FP/P(100,100),E(1000,2),A(50,2),DISMAX
C COMMON/G/LP(50),H(50),NP,LE,INFOR,OUT
C COMMON/H/IE(1000),IA(50)
C COMMON/GIRIA/NGIR,NNOD
C
C LOGICAL UTRD,SICLOC
C INTEGER OUT,P,E,A,DISMAX
C
C IF(IE(1).NE.1.
C   .AND.IE(N).NE.LE) GOTO 2
C
C DO 1 I=1,N-1
C   IN=N-(I-1)
C   IF(IE(IN-1).EQ.IE(IN)-1) GOTO 1
C
C   NGIR=I
C   NNOD=N
C
C   CALL CLOCK(I,N)
C
C   SICLOC=.TRUE.
C
C   GOTO 2
C
C 1 CONTINUE
C
C TEST DE 'CONTIGUIDAD'.
C
C 2 CONTINUE
C   DO 3 I=1,N-1
C     IF(IE(I).NE.IE(I+1)-1) GOTO 4
C 3 CONTINUE
C   RETURN
C 4 CONTINUE
C   UTRD=.TRUE.
C   RETURN
C
C *****
C
C END

```

```

C
C *****
C * ESTA SUBROUTINA SITUA UN LITERAL DE HASTA 240 CARACTERES -IN- *
C *CLUIDOS BLANCOS-, QUE LEE DE TARJETAS, SOBRE UN RECTANGULO DE OI- *
C *MENSIONES: XL*YL, CUYO ANGULO INFERIOR IZQUIERDO SE SITUA EN *
C * (XO,YO). *
C * ESTA SUBROUTINA NO ALTERA EL ORIGEN DE COORDENADAS DEFINIDO *
C *ANTES DE LLAMARLA. *
C * EL ROTULO ESTA FORMADO POR IMAGENES DE TARJETA. *
C *****
C
C SUBROUTINE TITULO(XO,YO,XL,YL)
C
C *****
C CHARACTER ROT(240)
C
C ROT = VECTOR DE CARACTERES.
C
C *****
C LEE EL NUMERO DE TARJETAS QUE FORMAN EL ENCABEZAMIENTO. MAXIMO:3.
C
C READ(5,100) N
C 100 FORMAT(I3)
C
C LEE LAS TARJETAS.
C
C LROT=80
C IF(N.E0.2) LROT=160
C IF(N.E0.3) LROT=240
C READ(5,101) (ROT(I),I=1,LROT)
C 101 FORMAT(80A1)
C
C *****
C SE TRASLADA EL ORIGEN DE COORDENADAS A (XO,YO) Y SE CALCULA EL TA-
C MA"O DE LOS CARACTERES DEL ROTULO -TAMX- PARA QUE LA REPRESENTA-
C CION SEA IDONEA.
C
C CALL PLOT(XO,YO,-3)
C ORXROT=1.
C TAMX=(XL-2.)/80.
C TAMY=3*TAMX
C IF(N.E0.2) TAMY=5*TAMX
C IF(N.E0.3) TAMY=7*TAMX
C IF(TAMY.LE.YL) GOTO 10
C TAMY=YL
C TAMX=TAMY/3.
C IF(N.E0.2) TAMX=TAMY/5.
C IF(N.E0.3) TAMX=TAMY/7.
C ORXROT=(XL-TAMX*80.)/2.
C 10 CONTINUE
C
C *****
C SE ROTULA.
C

```

```
IPUNT=1
DO 1 I=1,N
ORYROT=YL-2*I*TAMX
CALL SYMBOL(ORXROT,ORYROT,TAMX,ROT(IPUNT),0.,80)
IPUNT=IPUNT+80
1 CONTINUE
C
C*****
C
C SE DESPLAZA EL ORIGEN DE COORDENADAS A DONDE ESTABA ANTES DE LLAMAR
C A ESTA SUBROUTINA.
C
C CALL PLOT(-XD,-YD,-3)
C*****
C
C
C RETURN
C END
```

```
C
C .....
C * SUBROUTINA DE TRAMADO DE RECTANGULOS. *
C .....
C
SUBROUTINE TRAMA(J)
COMMON/E/XS(4,180),YS(4,180)
IF(ABS(XS(3,J)).GT.ABS(XS(1,J))) GOTO 1
CALL TRAMEL(J,4,90.,0.07,1)
GOTO 2
1 CONTINUE
CALL TRAMEL(J,4,90.,0.02,1)
2 CONTINUE
C
C .....
C
RETURN
END
```

```

C
C *****
C *   ESTA SUBROUTINA SITUA SOBRE EL POLIGONO DEFINIDO POR LAS CDR- *
C *DENADAS DE SUS VERTICES ALMACENADAS EN XY, LA TRAMA IPUNT, DE LA *
C *TABLA DE TRAMAS QUE SE DFRECE AL USUARIO. *
C *****
C
C   SUBROUTINE TRAMA(XY,MAX,M,IPUNT)
C
C   MAX = DIMENSION DE XY EN EL PROGRAMA DE LLAMADA A ESTA SUBROUTINA.
C   M = NUMERO DE PUNTOS QUE DESCRIBEN EL POLIGONO.
C *****
C
C   COMMON/H/NRAY(22),TRAMAS(8,22)
C   DIMENSION XY(MAX,2)
C
C   EN LA ZONA COMUN H ESTA ALMACENADA LA DESCRIPCION NUMERICA DE LAS
C   TRAMAS.
C *****
C
C   DATA INFOR/1/
C   N=NRAY(IPUNT)
C   DO 1 I=1,N
C   I2=I*2
C   I2M1=I2-1
C   IF(INFOR.NE.1)
C *WRITE(6,100) I,TRAMAS(I2M1,IPUNT),TRAMAS(I2,IPUNT)
100  FORMAT(I4,2F5.2)
C   CALL TRAMEL(XY,MAX,M,TRAMAS(I2M1,IPUNT),TRAMAS(I2,IPUNT),1)
C   1 CONTINUE
C   RETURN
C *****
C
C   END

```

```

C
C *****
C * ESTA SUBROUTINA SITUA UN HAZ DE PARALELAS, QUE FORMAN UN ANGULO *
C *W CON EL EJE POSITIVO DE LAS X, CUYA SEPARACION ENTRE LINEAS ES *
C *DR, SOBRE UN POLIGONO DEFINIDO POR LAS COORDENADAS DE SUS VERTI- *
C *CES, ALMACENADAS EN LA ZONA COMUN E. *
C *****
C
SUBROUTINE TRAMEL(IJ,LXY,W,DR,INFOR)
COMMON/E/XS(4,180),YS(4,180)
DIMENSION XYA(50,2),A(20)
INTEGER OUT
DATA OUT/6/
YMIN=1.E74
YMAX=-1.E74
C
C *****
C SE ENMARCA EL POLIGONO A TRAMAR.
C
DO 1 I=1,LXY
CALL G1(XYA(I,1),XYA(I,2),XS(I,IJ),YS(I,IJ),W)
IF(YMAX.LT.XYA(I,2))YMAX=XYA(I,2)
IF(YMIN.GT.XYA(I,2))YMIN=XYA(I,2)
1 CONTINUE
IF(INFOR.NE.1)
WRITE(OUT,100)YMIN,YMAX
100 FORMAT(4F10,2)
C
C *****
C SE TRAMA EL POLIGONO.
C
M=(YMAX-YMIN)/DR+4.5
YR=YMIN-(DR+0.06789)
DO 20 J=1,M
YR=YR+DR
CALL RAY(XYA(1,1),XYA(1,2),LXY,YR,A,K)
IF(K.LE.0)GOTO 20
A(K+1)=999999.
IF(MOD(K,2).NE.0)K=K+1
DO 2 I=1,K,2
CALL G2(XN,YN,A(I),YR,W)
CALL PLOT(XN,YN,3)
CALL G2(XN,YN,A(I+1),YR,W)
CALL PLOT(XN,YN,2)
2 CONTINUE
20 CONTINUE
C
C *****
C
RETURN
END

```

```

C
C
C *****
C *  ESTA SUBROUTINA SITUA UN HAZ DE PARALELAS, QUE FORMAN UN ANGULO *
C *W CON EL EJE POSITIVO DE LAS X, CUYA SEPARACION ENTRE LINEAS ES *
C *DR, SOBRE UN POLIGONO DEFINIDO POR LAS COORDENADAS DE SUS VERTI- *
C *CES, ALMACENADAS EN XY. *
C *****
C
C   SUBROUTINE TRAMEL(XY,MAX,LXY,W,DR,INFOR)
C   DIMENSION XY(MAX,2),XYA(50,2),A(20)
C   INTEGER OUT
C   DATA OUT/6/
C   YMIN=1.E74
C   YMAX=-1.E74
C
C *****
C
C   SE ENMARCA EL POLIGONO A TRAMAR.
C
C   DO 1 I=1,LXY
C   CALL G1(XYA(I,1),XYA(I,2),XY(I,1),XY(I,2),W)
C   IF(YMAX.LT.XYA(I,2))YMAX=XYA(I,2)
C   IF(YMIN.GT.XYA(I,2))YMIN=XYA(I,2)
C   1 CONTINUE
C   IF(INFOR.NE.1)
C   *WRITE(OUT,100)YMIN,YMAX
C 100 FORMAT(4F10.2)
C
C *****
C
C   SE TRAMA EL POLIGONO.
C
C   M=(YMAX-YMIN)/DR+4.5
C   YR=YMIN-(DR+0.06789)
C   DO 20 J=1,M
C   YR=YR+DR
C   CALL RAY(XYA(1,1),XYA(1,2),LXY,YR,A,K)
C   IF(K.LE.0)GOTO 20
C   A(K+1)=999999.
C   IF(MOD(K,2).NE.0)K=K+1
C   DO 2 I=1,K,2
C   CALL G2(XN,YN,A(I),YR,W)
C   CALL PLOT(XN,YN,3)
C   CALL G2(XN,YN,A(I+1),YR,W)
C   CALL PLOT(XN,YN,2)
C   2 CONTINUE
C 20 CONTINUE
C
C *****
C
C   RETURN
C   END

```

```

C
C *****
C * ESTA SUBROUTINA PREPARA LA INFORMACION NECESARIA PARA DIBUJAR LA *
C * LEYENDA Y EL DIAGRAMA DE FRECUENCIAS. *
C *****
C
C SUBROUTINE TRAMLY(N,ISW)
C
C N = NUMERO DE CASOS DEL PROBLEMA DE DIBUJO.
C ISW = BIFURCADOR. SI ES IGUAL A 1, PREPARA LA INFORMACION NECESARIA
C PARA LA LEYENDA Y EL DIAGRAMA. SI ES IGUAL A 2 SOLAMENTE LA
C CORRESPONDIENTE AL DIAGRAMA.
C *****
C
C COMMON/B/ITRAM(100)
C COMMON/D/ICLAS,IT(10)
C COMMON/E/UMB(9)
C COMMON/F/IFRECT(10,2),UMBIS(10,2)
C
C IFRECT(1,1) = CODIGOS DE LAS TRAMAS.
C IFRECT(1,2) = FRECUENCIAS DE CADA UNA DE LAS TRAMAS.
C UMBIS(1,1) = UMBRAL INFERIOR DE CADA CLASE.
C UMBIS(1,2) = UMBRAL SUPERIOR DE CADA CLASE.
C *****
C
C SE CUENTA EL NUMERO DE VECES QUE SE REPITE UNA TRAMA Y SE CREA LA
C LISTA DE FRECUENCIAS Y, COHERENTE CON ELLA, LA LISTA DE UMBRALES
C SUPERIOR EN INFERIOR ASOCIADOS A CADA TRAMA.
C
C DG 1 I=1,ICLAS
C MEM = IT(I)
C IFRC=0
C DO 2 J=1,N
C IF(ITRAM(J).NE.MEM) GOTO 2
C IFRC=IFRC+1
C 2 CONTINUE
C IFRECT(I,1)=MEM
C IFRECT(I,2)=IFRC
C IF(ISW.NE.1) GOTO 1
C UMBIS(I,1)=UMB(I-1)
C UMBIS(I,2)=UMB(I)
C IF(I.EQ.1.OR.I.EQ.ICLAS) GOTO 3
C GOTO 1
C 3 CONTINUE
C IF(I.EQ.1) UMBIS(I,1)=0.
C IF(I.EQ.ICLAS) UMBIS(I,2)=0.
C 1 CONTINUE
C *****
C
C LA ZONA COMUN F YA ESTA PREPARADA PARA QUE PUEDAN ENTRAR EN ELLAS
C LAS SUBROUTINAS QUE DIBUJAN EL HISTOGRAMA Y LA LEYENDA.
C
C RETURN
C *****

```



) C  
) END

```
C
C *****
C * SUBROUTINA FALSA DE TRANSFORMACION DE DATOS. *
C *****
C
SUBROUTINE TRANFR(PM,PF,N,DATMA,DATFA)
DIMENSION PM(N),PF(N),DATMA(N),DATFA(N)
DO 2 I=1,N
DATMA(I)=PM(I)
DATFA(I)=PF(I)
2 CONTINUE
DO 1 I=1,N
PM(I)=PM(I)
PF(I)=PF(I)
1 PF(I)=PF(I)
C
C *****
C
RETURN
END
```

```
C
C *****
C * SUBROUTINA FALSA DE TRANSFORMACION DE DATOS. *
C *****
C
C SUBROUTINE TRANF2(PM,PF,N,DATMA,DATFA)
C DIMENSION PM(N),PF(N),DATMA(N),DATFA(N)
C DO 1 I=1,N
C   PM(I)=PM(I)
C 1 PF(I)=PF(I)
C DO 2 I=1,N
C   DATMA(I)=DATMA(I)
C 2 DATFA(I)=DATFA(I)
C *****
C
C RETURN
C END
```

```

C
C *****
C * SUBROUTINA DE TRANSFORMACION DE LOS DATOS APORTADOS POR EL USUA-
C * RIO, EN FUNCION DE SUS PETICIONES, MANIFESTADAS POR UNA SERIE DE *
C * PARAMETROS QUE SE LEEN EN ESTA SUBROUTINA.
C *****
C
C SUBROUTINE TRANSF(N)
C
C N = NUMERO DE PIRAMIDES A TRANSFORMAR.
C *****
C
C COMMON/A/DATOS(90,40),ICOD(20),NGE(20),IAG(20),IAGF(20)
C COMMON/B/ROTUL(80,21)
C DIMENSION DATMA(90),DATFA(90)
C INTEGER OUT
C CHARACTER ROTUL
C
C TODOS LOS ARRAYS QUE FORMAN PARTE DE LAS ZONAS COMUNES A Y B YA HAN
C SIDO DESCRITOS EN EL PROGRAMA PRINCIPAL.
C DATMA = VECTOR AUXILIAR PARA TRANSFORMACION DE POBLACIONES MASCULI-
C NAS.
C DATFA = VECTOR AUXILIAR PARA TRANSFORMACION DE POBLACIONES FEMENI-
C NAS.
C *****
C
C INICIALIZACION DE LAS VARIABLES DE LECTURA Y ESCRITURA.
C
C DATA IN,OUT/7,6/
C
C LA INFORMACION REFERENTE A LAS TRANSFORMACIONES QUE HAY QUE HACER
C CON LOS DATOS DEL USUARIO FORMA PARTE DEL FICHERO FT07F001.
C *****
C
C LECTURA DE PARAMETROS DE LAS TRANSFORMACIONES.
C
C NM1=N+1
C READ(IN,100) (ROTUL(I,NM1),I=1,18),IAGR,I TRAN
C 100 FORMAT(18A1,12,T29,12)
C WRITE(OUT,101) (ROTUL(I,NM1),I=1,18),IAGR,I TRAN
C 101 FORMAT(1X,18A1,12,T30,12)
C *****
C
C LECTURA DE LAS AGREGACIONES DESEADAS Y REALIZACION DE LAS TRANSFOR-
C MACIONES SOLICITADAS.
C
C DO 1 I=1,N
C 1 IAGF(I)=5
C DO 2 K=1,N
C K2=K*2
C K2M1=K2-1
C IF(IAGR.E0.1) GOTO 4
C IF(IAGR.E0.2.AND.K.GT.1) GOTO 3
C READ(IN,102) (ROTUL(I,NM1),I=1,35),IAGF(K)

```

```

102 FORMAT(35A1,12)
WRITE(OUT,103) (ROTUL(I,NM1),I=1,35),IAGF(K)
103 FORMAT(1X,35A1,12,' A"OS')
GOTO 4
3 CONTINUE
IAGF(K)=IAGF(K-1)
4 CONTINUE
CALL AGREGA(DATOS(1,K2M1),DATOS(1,K2),NGE(K),90,IAGF(K),IAG(K),
*NL)
NGE(K)=NL
IF(1COD(K),NE,2) GOTO 7
IF(1TRAN.E0,2) GOTO 5
CALL PORCE2(DATOS(1,K2M1),DATOS(1,K2),NGE(K),DATMA,DATFA)
GOTO 9
5 CALL TRANF2(DATOS(1,K2M1),DATOS(1,K2),NGE(K),DATMA,DATFA)
GOTO 9
7 CONTINUE
IF(1TRAN.E0,2) GOTO 8
CALL PORCEN(DATOS(1,K2M1),DATOS(1,K2),NGE(K),DATMA,DATFA)
GOTO 9
8 CALL TRANFR(DATOS(1,K2M1),DATOS(1,K2),NGE(K),DATMA,DATFA)
9 CONTINUE
2 CONTINUE
C
C.....
C
C LLAMADA A LA SUBROUTINA DE DIBUJO.
C
C CALL DIBUJO(N)
C
C.....
C
C RETURN
C END

```

```
C
C *****
C * SUBROUTINA FALSA PARA LECTURA O TRANSFORMACION DE DATOS QUE SE *
C * REFIEREN AL PESO DE UNA VARIABLE SOBRE LAS AREAS DEL PROBLEMA DE *
C * DIBUJO. *
C *****
C
C SUBROUTINE TRANS1(N)
C COMMON/A/P(100)
C N=N
C RETURN
C *****
C
C END
```

```
C
C *****
C * SUBROUTINA FALSA PARA LECTURA O TRANSFORMACION DE DATOS QUE SE *
C * REFIEREN A LOS PESOS DE DIVERSAS VARIABLES (MAXIMO 10) SOBRE LAS *
C * AREAS DEL PROBLEMA DE DIBUJO, O A LAS DISTANCIAS INTERCASOS, CAL- *
C * CULADAS PREVIAMENTE. *
C *****
C
C SUBROUTINE TRANS2(N,NVAR)
C COMMON/C/PS(100,10)
C COMMON/G/D(100,100)
C N=N
C NVAR=NVAR
C RETURN
C *****
C
C END
```

```

C
C *****
C * SUBROUTINA QUE DIBUJA UN DIAGRAMA TRIANGULAR. *
C *****
C
C SUBROUTINE TRIANG(DATOS,N,IAMPL,IOPT,FCT)
C
C DATOS = MATRIZ DE DIMENSION 3*N DONDE SE ALMACENAN LOS % DE LOS
C TRES GRANDES GRUPOS DE EDAD. CADA COLUMNA REPRESENTA UN CA-
C SO. LA PRIMERA FILA SON LOS JOVENES, LA SEGUNDA LOS ADULTOS,
C LA TERCERA LOS VIEJOS.
C N = NUMERO DE CASOS DEL PROBLEMA DE DIBUJO.
C IAMPL = PARAMETRO QUE CONTROLA EL QUE SE DIBUJE TODO EL DIAGRAMA
C O SOLO LA PARTE AFECTADA POR LOS CASOS DEL PROBLEMA.
C IOPT = PARAMETRO QUE CONTROLA EL QUE APAREZCAN MOTAS O NUMEROS.
C FCT = FACTOR DE ESCALA DEL DIBUJO.
C *****
C
C DIMENSION DATOS(3,N)
C DIMENSION VINDIC(2),VINDI1(2)
C REAL L,L1
C INTEGER OUT
C
C VINDIC Y VINDI1 = POSICIONES DE MEMORIA DONDE SE ALMACENAN LAS COM-
C PONENTES DE LOS VECTORES UNITARIOS DEL EJE 'JOVE-
C NES' Y DEL EJE 'ADULTOS' DEL DIAGRAMA.
C *****
C
C INICIALIZACION DE VARIABLES.
C
C DATA OUT/6/
C DATA INFOR/2/
C DATA L,TAM,TAM2/20.,0.2,0.1/
C DATA DEJ,DEA,DEV/0.,0.,0./
C *****
C
C SI EL PROGRAMA PASA EL CONTROL A LA INSTRUCCION ETIQUETADA CON EL
C NUMERO 200 SE DIBUJA TODO EL DIAGRAMA.
C
C IF(IAMPL.EQ.0) GOTO 200
C *****
C
C SE LOCALIZAN LOS MAYORES MULTIPLOS DE 10, IGUALES O INFERIORES A
C LOS MINIMOS DE CADA FILA DE LA MATRIZ DE DATOS. ESTO DEFINIRA LOS
C VALORES DE LOS VERTICES DEL TRIANGULO.
C
C IF(INFOR.NE.1)
C *WRITE(OUT,101)(J,(DATOS(I,J),I=1,3),J=1,N)
101 FORMAT(13,3F10.2)
C RMINJ=DATOS(1,1)
C RMINA=DATOS(2,1)
C RMINV=DATOS(3,1)
C DO1 I=2,N
C IF(DATOS(1,I).LT.RMINJ)RMINJ=DATOS(1,I)

```



```

      IF(DATOS(2,1).LT.RMINA)RMINA=DATOS(2,1)
      IF(DATOS(3,1).LT.RMINV)RMINV=DATOS(3,1)
1    CONTINUE
      IF(INFOR.NE.1)
      *WRITE(OUT,100)RMINJ,RMINA,RMINV
100  FORMAT(5X,3F7.2)
      IJ=RMINJ/10.
      IA=RMINA/10.
      IV=RMINV/10.
      DEJ=IJ*10
      DEA=IA*10
      DEV=IV*10
C
C*****
C
200 CONTINUE
      IF(INFOR.NE.1)
      *WRITE(OUT,100) DEJ,DEA,DEV
      HASTAJ=100.-(DEA+DEV)
      HASTAA=100.-(DEJ+DEV)
      HASTAV=100.-(DEJ+DEA)
      IF(INFOR.NE.1)
      *WRITE(OUT,100) HASTAJ,HASTAA,HASTAV
C
C*****
C
      SECCION DE DIBUJO.
      CALL PLOTS(0,0,6)
      CALL FACTOR(FCT)
C
C SE PINTA UN TRIANGULO EQUILATERO DE DIMENSIONES
C PROPORCIONADAS A LAS DEL CUADRO, QUE TAMBIEN SE DIBUJA.
C
      A=4.
      L1=L-2*A
      CALL PLOT(A,A,3)
      CALL PLOT(A+L1,A,2)
      CALL PLDT(A+L1/2.,A+L1/2.*SQRT(3.),2)
      CALL PLOT(A,A,2)
C
C SI EL PROGRAMA PASA EL CONTROL A LA INSTRUCCION ETIQUETADA CON EL
C NUMERO 202 SE VAN A SITUAR NUMEROS EN EL INTERIOR DEL DIAGRAMA PARA
C OBTENER UN SUPERPONIBLE. NO SE ROTULAN LOS EJES NI LAS DIVISIONES
C INTERNAS DEL DIAGRAMA.
C
      IF(IOPT.NE.0) GOTO 202
C
C*****
C
C SE ROTULAN COHERENTEMENTE LOS LADOS DEL TRIANGULO Y SE DIBUJAN
C LAS DIVISIONES INTERIORES, PARA FACILITAR LA LECTURA DEL GRAFICO.
C
      AMPLI=(HASTAJ-DEJ)/10.
      IF(IAMPL.NE.0) AMPLI=(HASTAJ-DEJ)/5.
      SEP=L1/AMPLI
      IAMPL=INT(AMPLI+1.)
      Y=A

```

```

ROTULD=10.
IF(IAMPL.NE.0)ROTULD=5.
DO 6 I=1,IAMPL1
X=A+SEP*(I-1)
ROTULJ=DEJ+ROTULD*(I-1)
ROTULA=HASTAA-ROTULD*(I-1)
ROTULV=DEV+ROTULD*(I-1)
IF((I.EQ.1).OR.(I.EQ.IAMPL1))GOTO 7
CALL NUMBER(X-TAM,Y-2*TAM,TAM,ROTULJ,0.,-1)
CALL PLOT(X,Y,3)
CALL INTERS(X,Y,SQRT(3.),A+L1,A,A+L1/2.,A+L1/2.*SQRT(3.),X1,Y1)
CALL PLOT(X1,Y1,2)
CALL NUMBER(X1+2*TAM,Y1-(TAM/2.),TAM,ROTULA,0.,-1)
CALL PLOT(X1,Y1,3)
CALL INTERS(X1,Y1,0.,A+L1/2.,A+L1/2.*SQRT(3.),A,A,X2,Y2)
CALL PLOT(X2,Y2,2)
CALL NUMBER(X2-3*TAM,Y2-(TAM/2.),TAM,ROTULV,0.,-1)
CALL PLOT(X2,Y2,3)
CALL PLOT((A+L1-SEP*(I-1)),Y,2)
GOTO 6
7 CONTINUE
IF(I.EQ.IAMPL1)GOTO 8
CALL NUMBER(X-TAM,Y-2*TAM,TAM,ROTULJ,0.,-1)
CALL NUMBER(A+L1/2.+TAM,A+L1/2.*SQRT(3.)-TAM/2.,TAM,ROTULA,0.,-1)
CALL NUMBER(A+L1/2.-3*TAM,A+L1/2.*SQRT(3.)-TAM/2.,TAM,ROTULV,0.,-1
*)
GOTO 6
8 CONTINUE
CALL NUMBER(X-TAM,Y-2*TAM,TAM,ROTULJ,0.,-1)
CALL NUMBER(X+TAM,Y-TAM/2.,TAM,ROTULA,0.,-1)
CALL NUMBER(A-3*TAM,Y-TAM/2.,TAM,ROTULV,0.,-1)
6 CONTINUE
C
C*****
C
C 202 CONTINUE
C
C SE DIBUJAN LAS LEYENDAS DE LOS TRES EJES(LADOS)DEL TRIANGULO.
C
C TAM1=TAM*3.
C X1=A+L1/2.-3*TAM1
C Y1=A/2.-TAM1/2.
C X2=L-8*TAM1
C Y2=A+(L1/2.*SQRT(3.))/2.-TAM1/2.
C X3=TAM1
C CALL SYMBOL(X1,Y1,TAM1,'JOVENES',0.,7)
C CALL SYMBOL(X2,Y2,TAM1,'ADULTOS',0.,7)
C CALL SYMBOL(X3,Y2,TAM1,'VIEJOS',0.,6)
C
C SE LLEVA EL ORIGEN AL VERTICE INFERIOR IZQUIERDO DEL TRIANGULO.
C
C CALL PLOT(A,A,-3)
C
C SE DEFINEN LOS PARAMETROS NECESARIOS PARA PINTAR LOS PUNTOS.
C
C OR1=DEJ
C OR2=DEA
C ESCAL=L1/(HASTAJ-DEJ)

```

```

VINDIC(1)=1.
VINDIC(2)=0.
VINDI1(1)=1./2.
VINDI1(2)=SQRT(3.)/2.
R=0.1
C
C SE PINTAN TODOS LOS PUNTOS.
C
YD=TAM2/2.
DO 13 J=1,N
X=(DATOS(1,J)-OR1)*ESCAL+VINDIC(1)+(DATOS(2,J)-OR2)*ESCAL
**VINDI1(1)
Y=(DATOS(1,J)-OR1)*ESCAL+VINDIC(2)+(DATOS(2,J)-OR2)*ESCAL
**VINDI1(2)
IF(IOPT.EQ.0)GOTO 201
XD=TAM2/2.
IF(J.GE.10)XD=TAM2
XN=X-XD
YN=Y-YD
RJ=FLDAT(J)
CALL NUMBER(XN,YN,TAM2,RJ,0.,-1)
GOTO 13
201 CALL MDTA(X,Y,R)
13 CONTINUE
C
C SE DEVUELVE EL ORIGEN DE COORDENADAS AL VERTICE INFERIOR IZQUIERDO.
C
CALL PLOT(-A,-A,-3)
C
C LLAMADAS A TITULO Y A MARCO.
C
XTIT=0.
CALL TITULO(XTIT,L-L/4.,L,L/4.)
XMARC=0.
YMARC=0.
CALL MARCO(XMARC,YMARC,L,L)
C
C SE CIERRA EL DIBUJO.
C
CALL PLOT(L+1.,L+1.,999)
C
C*****
C
RETURN
END

```

```
C
C *****
C * ESTA SUBROUTINA CALCULA LAS COORDENADAS EN EL ESPACIO IMAGEN DE *
C *UN PUNTO EN TRES DIMENSIONES. *
C *****
C
C SUBROUTINE TRN32 (T,R)
C
C T = COORDENADAS DEL PUNTO EN EL ESPACIO IMAGEN (X,Y).
C R = COORDENADAS DEL PUNTO EN EL ESPACIO ORIGEN (X,Y,Z).
C *****
C
C COMMON/A/A(3,3)/B/ITP/C/DR
C DIMENSION T(2),R(3)
C
C A = MATRIZ DE TRANSFORMACION CALCULADA EN VISTA.
C *****
C
C T(1)=R(1)*A(1,2)+R(2)*A(2,2)+R(3)*A(3,2)
C T(2)=R(1)*A(1,3)+R(2)*A(2,3)+R(3)*A(3,3)
C RETURN
C *****
C
C END
```

```

C
C *****
C *   ESTA SUBROUTINA CALCULA LA MATRIZ DE UNA TRANSFORMACION   *
C *QUE LLEVA EL EJE X SOBRE EL PUNTO DE VISTA (X0,Y0,Z0),      *
C *MEDIANTE DOS GIROS SUCEIVOS, PRIMERO EN TORNO AL EJE Z      *
C *Y LUEGO EN TORNO AL EJE Y GIRADO.                            *
C *****
C
C   SUBROUTINE VISTA(X0,Y0,Z0)
C
C   X0 = COORDENADA X DEL PUNTO DE VISTA.
C   Y0 = COORDENADA Y DEL PUNTO DE VISTA.
C   Z0 = COORDENADA Z DEL PUNTO DE VISTA.
C *****
C
C   COMMON/A/A(3,3)
C   DIMENSION B(3,3),C(3,3)
C
C   A = MATRIZ DE TRANSFORMACION.
C   B Y C = MATRICES AUXILIARES.
C *****
C
C   CALCULO DE LOS ANGULOS DE GIRO.
C   W1=ATAN2(Y0,X0)
C   W2=ATAN2(Z0,SQRT(X0*X0+Y0*Y0))
C
C   GIRO EN TORNO AL EJE Z.
C   CW=COS(W1)
C   SW=SIN(W1)
C   C(1,1)=CW
C   C(1,2)=-SW
C   C(1,3)=0.
C   C(2,1)=SW
C   C(2,2)=CW
C   C(2,3)=0.
C   C(3,1)=0.
C   C(3,2)=0.
C   C(3,3)=1.
C
C   GIRO EN TORNO AL EJE Y.
C   CW=COS(W2)
C   SW=SIN(W2)
C   B(1,1)=CW
C   B(2,1)=0.
C   B(3,1)=SW
C   B(1,2)=0.
C   B(2,2)=1.
C   B(3,2)=0.
C   B(1,3)=-SW
C   B(2,3)=0.
C   B(3,3)=CW
C
C   CALCULO DE LA TRANSFORMACION COMPUESTA DE LOS DOS GIROS.
C   DO 5 I=1,3
C   DO 4 J=1,3
C   A(I,J)=0.

```

```
      DO 3 K=1,3
      A(I,J)=A(I,J)+C(I,K)*B(K,J)
3     CONTINUE
4     CONTINUE
5     CONTINUE
C
C  IMPRESION DE LAS MATRICES DE LAS TRANSFORMACIONES PARA VERIFICACION.
      WRITE(6,100) X0,Y0,Z0,A,B,C
100  FORMAT('1P. DE VISTA = ',3F10.3
      *//' A = '/3(6X,3F10.3/)
      *//' B = '/3(6X,3F10.3/)
      *//' C = '/3(6X,3F10.3/)
      RETURN
C
C .....
C
      END
```

```

C
C *****
C * SUBROUTINA DE INTERPOLACION MEDIANTE LA FORMULA: *
C * M M *
C * Z(I)=(SUMA(Z(K)/D(I,K)**2))/(SUMA(1/(D(I,K)**2))) *
C * K=1 K=1 *
C * DONDE: *
C * Z(I) = VALOR DE LA FUNCION EN UN PUNTO. *
C * M = NUMERO DE CENTROIDES. *
C * Z(K) = VALOR DE LA CARACTERISTICA EN EL CENTROIDE K. *
C * D(I,K) = DISTANCIA LINEAL ENTRE EL PUNTO Y EL CENTROIDE K. *
C *****
C
C SUBROUTINE ZETA(X,Y,Z,XV,YV,ZV,D,N)
C
C X = COORDENADA X DEL PUNTO.
C Y = COORDENADA Y DEL PUNTO.
C Z = VALOR DE LA FUNCION EN ESE PUNTO (LO CALCULA ESTA SUBROUTINA).
C XV = VECTOR DE COORDENADAS X DE TODOS LOS CENTROIDES.
C YV = VECTOR DE COORDENADAS Y DE TODOS LOS CENTROIDES.
C ZV = VECTOR DE VALORES DE LA CARACTERISTICA EN TODOS LOS CENTROIDES.
C D = VECTOR DE DISTANCIAS DESDE EL PUNTO A TODOS LOS CENTROIDES.
C N = NUMERO DE CENTROIDES.
C *****
C
C DIMENSION XV(N),YV(N),ZV(N),D(N)
C F(A,B)=A*A+B*B
C S=0
C DO 1 I=1,N
C D(I)=F(XV(I)-X,YV(I)-Y)
C D(I)=AMAX1(1.E-3,D(I))
C S=S+1./D(I)
C 1 CONTINUE
C Z=0
C DO 2 I=1,N
C Z=Z+ZV(I)/D(I)
C 2 CONTINUE
C Z=Z/S
C RETURN
C *****
C
C END

```

```

C
C *****
C * SUBROUTINA DE INTERPOLACION MEDIANTE UN POLINOMIO  $Z = F(X,Y)$  *
C * -SUPERFICIE DE TENDENCIA-, AJUSTADO A LOS VALORES DE LA DISTRIBU- *
C * CION DE UNA CARACTERISTICA EN UNA MUESTRA ESPACIAL. *
C *****
C
C SUBROUTINE ZETA2(X,Y,Z,N,A,N1)
C
C X = COORDENADA X DEL PUNTO.
C Y = COORDENADA Y DEL PUNTO.
C Z = VALOR DE LA FUNCION EN ESE PUNTO (LO CALCULA ESTA SUBROUTINA).
C N = GRADO DE LA SUPERFICIE + 1.
C A = MATRIZ EN LA QUE SE ENCUENTRAN ALMACENADOS LOS COEFICIENTES DEL
C POLINOMIO.
C N1 = DIMENSION DE LA MATRIZ A EN EL PROGRAMA QUE LLAMA A ESTA SUBRU-
C TINA.
C *****
C
C DIMENSION A(N1,N1)
C Z = 0.
C DO 1 I=1,N
C ZETA = 0.
C DO 2 J=1,I
C ZETA = ZETA + A(I,J)*X**(I-J)*Y**(J-1)
C 2 CONTINUE
C Z = Z + ZETA
C 1 CONTINUE
C RETURN
C *****
C
C END

```





398

G L O S A R I O

Array: Conjunto de posiciones de memoria que tienen todas un nombre común, distinguiéndose unas de otras por el subíndice que les corresponde. Por ejemplo, los  $n$  elementos del array LISTA se denominarán LISTA(1), LISTA(2)... LISTA(N). Los arrays pueden tener una, dos, tres o más dimensiones. Los arrays de una dimensión se conocen también por el nombre de vectores, y los de dos dimensiones por el de matrices. Para identificar todos los elementos de un vector tan sólo es necesario utilizar un subíndice. Para identificar los elementos de una matriz es necesario utilizar dos subíndices. En el ejemplo anterior, LISTA es un vector. Sea ahora una matriz de nombre LISTAS, de  $n \times m$  elementos, éstos se reconocerán por LISTAS(1,1), LISTAS(1,2)... LISTAS(N,M).

Biblioteca de programas: Conjunto de programas de ordenador que realizan tareas complementarias, estructurados de tal forma que su uso alternativo resulta muy sencillo y flexible, en base a una serie de instrucciones simples que aparecen descritas en el manual de uso de la biblioteca.

Dicomed: Transformador cinta/película. Este periférico impresiona una placa fotográfica, a partir de la descripción digital, grabada previamente en una cinta magnética, de una imagen.

Digitizador: Grabador automático de coordenadas. Recorriendo por puntos con el cursor de la máquina una determinada imagen, las coordenadas de éstos quedan automáticamente grabadas en un soporte informático, normalmente una cinta magnética.

Fichero: Aunque este término tiene diversas acepciones, en este trabajo siempre ha sido utilizado como sinónimo de conjunto de datos grabados en un soporte informático.

FORTAN: Lenguaje de programación de alto nivel desarrollado con una orientación hacia aplicaciones científicas. Se conocen muy diversas versiones del mismo. Los programas que se presentan en esta tesis han sido redactados en la versión FORTRAN77.

Generalización cartográfica: Conjunto de tareas implicadas en la selección adecuada de la información de base para la construcción de un mapa, en función de la escala de la representación.

Hardware: Término técnico importado del inglés con el que nos referimos a las bases físicas -máquinas implicadas- de un sistema informático.

Impresora: Periférico de impresión de caracteres alfanuméricos sobre papel, bajo control de un ordenador digital. Existen muy diversos tipos, en función de los diferentes procedimientos de impresión. Además, si la impresión se realiza carácter por carácter,

se habla de impresoras de caracteres, y si la impresión se lleva a cabo línea a línea de escritura, de impresoras de líneas. Estas últimas son mucho más veloces en cuanto al volumen total de información reproducida por unidad de tiempo.

JCL: Siglas de Job Control Language. Todo ordenador dispone de un conjunto de instrucciones de JCL, que posibilitan la ejecución de programas en lenguaje máquina y el procesado de información en su seno.

Lenguaje de programación: Gramática de comunicación con un ordenador digital, que permite llevar a cabo los objetivos de cálculo y tratamiento previstos en una determinada aplicación. Los primeros lenguajes de programación comprendían instrucciones muy simples, codificadas numéricamente, que se correspondían con las operaciones elementales que realiza electrónicamente un ordenador. Posteriormente se empezaron a desarrollar lenguajes de programación formados por instrucciones más próximas al lenguaje humano habitual -lenguajes de programación de alto nivel-, confiando a un programa traductor -compilador- la tarea de convertir esas instrucciones complejas en series de instrucciones elementales, ejecutables ya directamente por el ordenador.

Memoria: Sección del ordenador en la que se almacenan electrónicamente cantidades numéricas o caracteres alfanuméricos que van a ser objeto de posteriores manipulaciones o cálculos explicitados en las instrucciones de que consta un programa.

Pantalla de rayos catódicos: En todos los sistemas de reciente creación se utilizan pantallas de rayos catódicos para visualizar la información que se introduce en el ordenador y la que éste devuelve como resultado de algún objetivo de cálculo o manipulación de datos. Existen pantallas de rayos catódicos para visualización de caracteres alfanuméricos, pantallas para visualización de gráficos y pantallas que responden a ambas necesidades al mismo tiempo.

Periférico: Reciben este nombre genérico todos los componentes de un sistema informático conectados al ordenador propiamente dicho, que permiten un diálogo con él -entrada, almacenado externo y salida de información-: lectoras de tarjetas perforadas, impresoras, unidades de disco, unidades de cinta magnética, teletipos, plotters, etc.

Plotter: Periférico gráfico controlado por un ordenador que permite reproducir dibujos sobre papel automáticamente. Existen diversos tipos de plotters, pero el que ha sido contemplado en los programas que se presentan en esta tesis es el plotter de pluma. Este tipo de plotter reproduce dibujos dirigiendo -levantada (sin dibujar), o bajada (dibujando)- una pluma o un bolígrafo presurizado a puntos del papel, en base a las coordenadas de éstos, que le son transmitidas por el ordenador.

Procedimiento: Aunque existen otras acepciones técnicas de este

término, a lo largo de todo el texto hay que considerarlo como sinónimo de programa.

Programa: Conjunto de instrucciones -sentencias- redactadas en algún lenguaje informático que causan la realización de un objetivo de cálculo o manipulación de datos.

Scanner: Sensor electrónico que convierte automáticamente toda la información gráfica que existe en una determinada área en una matriz de códigos numéricos, relacionados con la intensidad de color en cada porción infinitesimal del área "barrida".

Software: Término técnico importado del inglés con el que nos referimos genéricamente tanto a las herramientas de programación que todo sistema ofrece, como a la tarea de programación en sí y a sus resultados.

Soporte: Base física de cualquier información codificada que puede interpretarse adecuadamente un periférico de un sistema informático: tarjetas perforadas, cintas magnéticas, discos magnéticos, tambores, cintas perforadas, etc.

Subprograma: Conjunto de instrucciones de manipulación o cálculo que puede ser invocado desde otro programa o subprograma.

Subrutina: Sinónimo de subprograma.

Trazador incremental: Plotter de pluma. (ver plotter).

Vector: Array de una dimensión (ver array).

INDICE DE FIGURAS

	<u>Págs.</u>
<u>Introducción</u>	
Fig. I.1. Flujo pregunta-respuesta en un sistema de información geográfica .....	18
<u>Capítulo 1.</u>	
Fig. 1.1. Ejes de coordenadas cartesianas .....	31
Fig. 1.2. Descripción de una poligonal cerrada por pares de coordenadas .....	31
Fig. 1.3. Descripción de una poligonal abierta por pares de coordenadas .....	32
Fig. 1.4. Parámetros de dibujo de una pirámide de población .....	42
Fig. 1.5. Dibujo de la pirámide (1) .....	44
Fig. 1.6. Dibujo de la pirámide (2) .....	45
Fig. 1.7. Dibujo de la pirámide (3) .....	46
Fig. 1.8. Dibujo de la pirámide (4) .....	46
Fig. 1.9. Almacenamiento en memoria de los efectivos masculinos y femeninos de cada grupo de edad.	47
Fig. 1.10. Dibujo de la pirámide (5) .....	48
Fig. 1.11. Dibujo de la pirámide (6) .....	49
Fig. 1.12. Dibujo de la pirámide (7) .....	49
Fig. 1.13. Pirámide de población compuesta .....	52
Fig. 1.14. Pirámide de población compuesta (según dos criterios de clasificación) .....	53
Fig. 1.15. Dibujo de la pirámide compuesta (1) .....	55
Fig. 1.16. Dibujo de la pirámide compuesta (2) .....	56
Fig. 1.17. Dibujo de la pirámide compuesta (3) .....	56
Fig. 1.18. Dibujo de la pirámide compuesta (4) .....	57
Fig. 1.19. Dibujo de la pirámide compuesta (5) .....	58
Fig. 1.20. Dibujo de la pirámide compuesta (6) .....	59
Fig. 1.21. Dibujo de dos pirámides superpuestas (1) ...	61

	<u>Págs.</u>
Fig. 1.22. Dibujo de dos pirámides superpuestas (2).....	61
Fig. 1.23. Dibujo de dos pirámides superpuestas (3).....	63
Fig. 1.24. Rótulos en la pirámide de población .....	64
Fig. 1.25. Leyenda del tramado de una pirámide .....	64
Fig. 1.26. Años de nacimiento de las diversas cohortes de una pirámide de población .....	65
Fig. 1.27. Dibujo automático de una pirámide de pobla- ción simple .....	76
Fig. 1.28. Dibujo automático de una pirámide de pobla- ción compuesta .....	77
Fig. 1.29. Dibujo automático de una pirámide de pobla- ción doblemente compuesta .....	78
Fig. 1.30. Dibujo automático de dos pirámides de pobla- ción superpuestas .....	79
Fig. 1.31. Dibujo de las divisiones internas de un dia- grama triangular .....	81
Fig. 1.32. Dibujo automático del diagrama triangular...	88
Fig. 1.33. Dibujo automático del diagrama triangular....	89
Fig. 1.34. Dibujo automático del diagrama triangular....	90
 <u>Capítulo 2.</u>	
Fig. 2.1. Descripción de áreas por poligonales cerradas	97
Fig. 2.2. Descripción de áreas por líneas de frontera..	98
Fig. 2.3. Descripción de contornos simples.....	99
Fig. 2.4. Líneas de frontera interáreas .....	100
Fig. 2.5. Establecimiento de la envolvente de dos polí- gonos contiguos (1) .....	118
Fig. 2.6. Establecimiento de la envolvente de dos polí- gonos contiguos (2) .....	118
Fig. 2.7. Establecimiento de la envolvente de dos polí- gonos contiguos (3) .....	119
Fig. 2.8. Intersecciones de una línea recta con una po- lígonaal .....	131
Fig. 2.9. Intersecciones de una línea recta con dos po- ligonales .....	133
Fig. 2.10. Intersecciones de una línea recta con dos po- ligonales concéntricas .....	134
Fig. 2.11. Intersecciones de un haz de paralelas con una polígonaal.....	135

	<u>Págs.</u>
Fig. 2.12. Combinación de giros y rayados en la obtención de tramas simples .....	137
Fig. 2.13. Estructura del programa COROMAP. ....	143
Fig. 2.14. Densidades de población. Provincias españolas 1981 (Clasific. en intervalos de igual amplitud) .....	153
Fig. 2.15. Densidades de población. Provincias españolas 1981. (Clasificación por unidades de desviación típica) .....	154
Fig. 2.16. Densidades de población. Provincias españolas 1981. (Clasificación por umbrales) .....	155
Fig. 2.17. Densidades de población. Provincias españolas 1981. (Clasificación por centiles) .....	156
Fig. 2.18. Grandes grupos de provincias españolas en función de la evolución de su densidad de población .....	157
 <u>Capítulo 3.</u>	
Fig. 3.1. Proyección de un punto en un espacio bidimensional (1) .....	162
Fig. 3.2. Proyección de un punto en un espacio bidimensional (2) .....	163
Fig. 3.3. Cálculo de coordenadas de un punto en un sistema girado .....	163
Fig. 3.4. Proyecciones en el plano (Y,Z) de puntos del espacio tridimensional (X,Y,Z). ....	165
Fig. 3.5. Posición del observador y ángulos de giro del sistema de referencias .....	166
Fig. 3.6. Rayado de un polígono en tres dimensiones(1). ....	169
Fig. 3.7. Rayado de un polígono en tres dimensiones(2). ....	169
Fig. 3.8. Rayado de un polígono en tres dimensiones(3). ....	170
Fig. 3.9. Rayado de un polígono en tres dimensiones(4). ....	171
Fig. 3.10. Rayado de un polígono en tres dimensiones(5). ....	172
Fig. 3.11. Rayado de un polígono en tres dimensiones(6). ....	173
Fig. 3.12. Rayado de un polígono en tres dimensiones(7). ....	173
Fig. 3.13. Tratamiento de las líneas ocultas (1) .....	175
Fig. 3.14. Tratamiento de las líneas ocultas (2) .....	176
Fig. 3.15. Tratamiento de las líneas ocultas(3) .....	176
Fig. 3.16. Datos necesarios para la elaboración de un bloque diagrama (1) .....	179



	<u>Págs.</u>
Fig. 3.17. Datos necesarios para la elaboración de un bloque diagrama (2) .....	179
Fig. 3.18. Densidades de población provinciales.1981....	191
Fig. 3.19. Densidades de población provinciales.1981....	192
Fig. 3.20. Densidades de población provinciales.1981....	193
Fig. 3.21. Preferencias espaciales en la España peninsular (Superficie de tendencia de grado 3).....	208
Fig. 3.22. Preferencias espaciales en la España peninsular (Superficie de tendencia de grado 4).....	209
Fig. 3.23. Preferencias espaciales en la España peninsular (Superficie de tendencia de grado 5).....	210
Fig. 3.24. Preferencias espaciales en la España peninsular (Superficie de tendencia de grado 6) ....	211
Fig. 3.25. Datos necesarios para la utilización del programa PRISMAS (1) .....	212
Fig. 3.26. Datos necesarios para la utilización del programa PRISMAS (2) .....	213
Fig. 3.27. Densidades de población provinciales. 1981. .	227
Fig. 3.28. Densidades de población provinciales. 1981. .	228
Fig. 3.29. Densidades de población provinciales. 1981. .	229
<u>Capítulo 4.</u>	
Fig. 4.1. Evolución temporal de un fenómeno (1) .....	233
Fig. 4.2. Evolución temporal de un fenómeno (2) .....	234
Fig. 4.3. Evolución temporal de un fenómeno (3) .....	234
Fig. 4.4. Evolución temporal de un fenómeno (4) .....	235
Fig. 4.5. Representación espacial de una variable (1) .	236
Fig. 4.6. Representación espacial de una variable (2)..	237
Fig. 4.7. Representación de la variación temporal de una característica espacial .....	238
Fig. 4.8. Distribución espacial de una característica en el momento T1 .....	245
Fig. 4.9. Distribución espacial de una característica en el momento T2 .....	246
Fig. 4.10. Interpolación temporal lineal (1) .....	246
Fig. 4.11. Interpolación temporal lineal (2) .....	247
Fig. 4.12. Interpolación temporal lineal (3) .....	248
Fig. 4.13. Evolución de las densidades de población provinciales en el periodo 1950-1960 .....	251-261

106

B I B L I O G R A F I A

1. ADAMS, T., (1979), Characteristics of a national topographical digital data base, Msc. dissertation, Univ. of Durham.
2. ANDERSON, S.E., (1972) "The scientific uses of Computer Animation", Proceedings of the National Microfilm Association, vol. XXI, pp. 11184-11189.
3. ANDRE, A., (1980), L'expression graphique: cartes et diagrammes, Paris, Ed. Masson.
4. ANSCOMBE, F.J., (1973) "Graphs in statistical analysis", American Statistician, 27, pp. 17-21.
5. BAKER, L., (1974), A selection of geographical computer programs, Geographical Papers, London School of Economics, 6.
6. BARREDO RISCO, E., (1979), Información Geográfica Automatizada. El SIGNA del Instituto Geográfico Nacional. Fuentes para la creación de Bases de Datos, Memoria de Licenciatura, Facultad de Geografía e Historia, U.C.M.
7. BASSETT, K.A., (1972), "Numerical methods for map analysis", Progress in Geography, vol. 4, pp. 217-54, London, Ed. Arnold.
8. BAXTER, R.G., (1973) "GINO, a general purpose graphic system for cartographic", J.U.N.N.D., 10.
9. BEGUIN, H. y THISSE, J.F., (1978), An Axiomatic Approach to Geographical Space, Geographical Analysis, Communication Université de Fribourg, 28 pp.
10. BELL, S.B.M. y BICKMORE, D.P., (1975), "Interactive Cartography at ECU: Regional Geography a la mode", Proceedings of the International Symposium on Computer-Assisted Cartography, Washington, pp. 303-17.
11. BENNETT, R.J. y WRIGLEY, N. (eds.), (1980), Quantitative Geography in Britain: Retrospect and Prospect, London, Routledge and Kegan Paul.
12. BERTIN, J., (1973), Sémiologie graphique, Paris, Gauthier-Villars, 431 pp.
13. BERTIN, J., (1977), La graphique et le traitement graphique de l'information, Paris, Flammarion, 227 pp.
14. BICKMORE, D.P., (1967), "The scope of automatic cartography", Proceedings, Geological Society of London, 1642, pp. 205-9.

15. BICKMORE, D.P., (1968), "Maps for the computer age", Geographical Magazine, 41, pp. 221-227.
16. BICKMORE, D.P., (1969), "Computers and Geology: automated mapping for the geologist", Geographical Magazine, 42.
17. BISCO, R.L. (ed.), (1970), Data bases, computers and the social sciences.
18. BORKO, H., (1962), Computer applications in the behavioural sciences.
19. BOSQUE MAUREL, J., CEBRIAN DE MIGUEL, J.A. y BOSQUE SENDRA, J., (1981), Precios del suelo en la ciudad de Madrid (1967-1981), Comunicación al VII Coloquio de Geografía, Pamplona.
20. BOSQUE MAUREL, J., (1982), Desequilibrios espaciales y problemática regional en la España actual, Ciclo de Conferencias sobre Desequilibrios Regionales, Zaragoza.
21. BOSQUE SENDRA, J. y CEBRIAN DE MIGUEL, J.A., (1980), Una aportación al desarrollo de un sistema de información geográfica: El SIGNA del Instituto Geográfico Nacional, Comunicación presentada a la II Conferencia Internacional de Bases de Datos en Humanidades y Ciencias Sociales, Madrid.
22. BOSQUE SENDRA, J., (1980), Las elecciones del periodo de la Reforma Política (1977-1979) en España y en la provincia de Granada, Tesis Doctoral, Facultad de Filosofía y Letras, Universidad de Granada.
23. BOSQUE SENDRA, J., (1981), De la Reforma a la Autonomía. Tres años de elecciones en la provincia de Granada, Secretariado de Publicaciones de la Universidad de Granada, (en prensa).
24. BOUILLE, F., (1974), Création et utilisation d'une banque des données d'une carte géologique, Comunicación presentada al 7th International Conference on Cartography, Madrid.
25. BOUILLE, F., (1977), Structuring Cartographic Data and Spatial Processes with the Hypergraph-Based Data Structure, Comunicación al M.I.T. Cambridge Mass. Advanced Study Symposium on Topographical Data Structures for Geographic Information Systems.
26. BOYLE, A.R. y KROLL, F.S., (1974), On line collection and manipulation of two and three dimensional topographic data with the new system at HANSA LUFTBILD, Comunicación presentada al 7th International Conference on Cartography, Madrid.
27. BRACONNE, S., (1977), Utilisation des notions de Textures et de Structures en vue d'une cartographie automatique des images de télédétection; état du logiciel correspondant, Service des Applications Nouvelles et de la Télédétection de l'Institut Géographique National, Saint-Mandé.

28. BRODLIE, K.W. (1980), Mathematical Methods in Computer Graphics and Design, London, Academic Press, 147 pp.
29. BROPHY, D.M., (1972), Automated Linear Generalization in Thematic Cartography, Msc. dissertation, Univ. of Wisconsin.
30. CAMPBELL, S.G., (1977), "Computer algorithms for spatial data", Area, 9-2, pp.106-8.
31. CAPEL, H. y URTEAGA, J.L., (1982), Las nuevas geografías, Madrid, Salvat, 64 pp.
32. CAUVIN, C. y RIMBERT, S., (1975), La lecture numérique des cartes thématiques, Fribourg, Editions Universitaires.
33. CAUVIN, C., REYMOND, H. y HIRSCH, J., (1980), Cartographie Informatisée et Géographie Humaine, Strasbourg, C.N.R.S. Laboratoire de Cartographie Thématique.
34. CEBRIAN DE MIGUEL, J.A. y BOSQUE SENDRA, J., (1982), "Un modelo multirregional para la proyección de la población de las provincias españolas", Anales de Geografía de la Universidad Complutense, Madrid, (en prensa).
35. CEBRIAN DE MIGUEL, J.A. y GARCIA FERRANDEZ, M., (1982), "Sistemas de Representación y Cartografía Temática. Evolución de las densidades de población de las provincias españolas en el periodo 1900-1981", Anales de Geografía de la Universidad Complutense, Madrid, (en prensa).
36. CEBRIAN DE MIGUEL, J.A., (1982), "Nuevos planteamientos en la organización de los Sistemas Cartográficos Nacionales. Euro-Carto I (Oxford, 13-16.XII.81)", Anales de Geografía de la Universidad Complutense, Madrid, (en prensa).
37. CHAPOT BLANQUET, M., (1974), "Comentarios a un programa de cartografía automática", Revista de Geografía, VIII, 1-2, pp.215-218.
38. CHORLEY, R.J. y HAGGETT, P., (1965), "Trend surface analysis in geographical research", Transactions of the Institute of British Geographers, 37, pp. 47-67.
39. COMITE FRANÇAIS DE CARTOGRAPHIE, (1975), Rapport national sur l'automatisation de la cartographie en France, Paris, Les éditions internationales, fascicule n°63.
40. CORNWELL, B. y ROBINSON, A., (1966), "Possibilities for Computer Animated Films in Cartography", Cartographic Journal, 3, pp.79-82.
41. COX, N.J., ALDRED, B.K. y RHIND, D.W., (1980), "Towards a Geographical Data Type". Geoprocessing, 1, pp. 217-29.

42. CRAIN, I.K., (1970), "Computer interpolation and contouring of twodimensional data: a review", Geoexploration, 8, pp. 71-86.
43. CRAWFORD, P. y MARKS, R., (1973), "The visual effects of geometric rotations on three-dimensional maps", Professional Geographer, 25(3), pp.233-8.
44. DAVIS, J.C. y MCCULLAGH, M., (1975), Display and Analysis of Spatial Data, New York, Wiley and Sons.
45. DAWSON, J. y UNWIN, D.J., (1976), Computing for Geographers, London, David and Charles, 362 pp.
46. DENEGRÉ, J., (1973), "Développements récents de l'automatisation appliquée aux cartes topographiques à moyenne échelle", Annuaire International de Cartographie, XIII.
47. DIAZ ALVAREZ, J.R., (1980), "Recension bibliográfica de Computer mapping for resource analysis", Paralelo 37, 4, pp. 214-215.
48. DICKINSON, G.C., (1977), Statistical mapping and the presentation of statistics, London, Arnold, 196 pp.
49. DOUGLAS, D., (1971), "VIEWBOLCK: A computer program for constructing perspective view block diagrams", Revue Géographique de Montreal.
50. DOUGLAS, D. y PEUCKER, T., (1973), "Algorithms for the reduction of the number of points required to represent a digitized line or its caricature", The Canadian Geographer, vol 10, n°2, pp.112-22.
51. DOUGLAS, D. (ed.), (1977), Applications of Geographic Information Processing, Ottawa, Univ, of Ottawa.
52. DYER, B.N. y WELLMAN, F., (1970), "Techniques and Applications of graph plotters", en GREEN, E. y PARLSLOW, D. (eds.) Computer graphics in management, London, Gower Press.
53. EVERITT, B.S., (1978), Graphical Techniques for multivariate data, London, Heinemann.
54. FAUS PUJOL, L. y CALVO PALACIOS, J.L., (1977), "Notas sobre la aplicación de unas técnicas de análisis espacial", Geographicalia.
55. FAUS PUJOL, L. y CALVO PALACIOS, J.L. (1977), "Aportación metodológica al análisis del espacio urbano mediante ordenador: Aplicación a Zaragoza", en Medio físico, Desarrollo Regional y Geografía. V Coloquio de Geografía, Granada, Secretariado de Publicaciones de la Universidad de Granada, pp. 353-358.
56. FERNANDEZ GARCIA, F. y MORENO JIMENEZ, A., (1981), Elaboración automática del mapa de isoyetas en un área montañosa del centro peninsular, Comunicación al VII Coloquio de Geografía, Pamplona.

57. FROLOV, Y.S., (1978), "Theoretical aspects of the Cartographic Research Method Leningrad University", American Geographical Society, march, pp. 151-160.
58. GACHTER, E. y KILCHEMANN, A., (1971), "Nuevos ejemplos de uso de métodos cuantitativos; ordenadores y trazadores en geografía y cartografía", Geographica, C.S.I.C., n° 2.
59. GAITS, M., (1969), "Thematic mapping by computer", Cartographic Journal, 6, pp. 50-68.
60. GARCIA BALLESTEROS, A., (1982), Crecimiento y problemas de la población mundial, Madrid, Salvat, 64 pp.
61. GARCIA-COUREL, J.M., (1972), "Mapas estadísticos formados en ordenador", Geographica, C.S.I.C., n° 2, pp. 97-106.
62. GARCIA-COUREL, J.M., (1978), Short notes and current applications and advantages of automated procedures in cartography, Comunicación al Seminario de Cartografía Automática de Nairobi, Kenya.
63. GARCIA-COUREL, J.M., (1979), Digitización y edición de información cartográfica, Madrid, Instituto Geográfico Nacional.
64. GARFINKEL, D., (1962), "Programmed methods for printer graphical output", Communications, Association for Computing Mathematics, 5, pp. 477-479.
65. GEOMATIQUE, Commission Permanente (1979), Groupe "Cartographie Assistée par Ordinateur". Comptes rendus à la Commission, Institut Géographique National, Service des Applications Nouvelles, Saint-Mandé.
66. GILMARTIN, P.P., (1981), "Influences of Map Context on Circle perception", A.A.A.G., vol 71, n° 2, pp. 253-258.
67. GNANADESIKAN, R., (1977), Methods for statistical data analysis of multivariate observations, New York, Wiley and Sons.
68. GRANDTEAN, J. y ARNOLD, K., (1978), "Cartes établies en moyen d'un traceur", Geographica Helvetica, n° 1.
69. GRIMMEAU, J., (1977), "Cartographie par Plages et Discontinuités Spatiales. Application à la Mobilité de la main d'oeuvre en Belgique", L'Espace Géographique, Paris, Doin ed.
70. GUERMOND, Y. y otros, (1981), "Microcomputer graphics in multivariate analysis", en BENNETT, R.J. (ed.), European Progress in Spatial Analysis, London, Pion, pp. 268-279.
71. HACKATHORN, H., (1977), "ANIMA II: A 3-D Colour Animation System", Computer Graphics, 11, pp. 54-64.

72. HAGERSTRAND, T., (1967), "The geographer and the computer", Transactions of the Institute of British Geographers, 42, pp. 1-19.
73. HARALICK, R.M., (1980), "A Spatial Data Structure for Geographic Information Systems", en Map Data Processing, London, Academic Press.
74. HEAP, B.R. y PINK, M.G., (1969), "Three contouring algorithms", National Physical Laboratory, DNAM, 81.
75. HELLER, M., (1974), On the use of Automatic Photosetters in the production of thematic maps, Comunicación al 7th International Conference on Cartography, Madrid.
76. HERRERO GARCIA, R., BOSQUE SENDRA, J. y CEBRIAN DE MIGUEL, J.A., (1980); "Nuevas tendencias en la investigación geográfica: el sistema de información del Instituto Geográfico Nacional (SIG-NA)", Estudios Geográficos, nº 161, pp. 447-465.
77. HERRERO GARCIA, R. y CEBRIAN DE MIGUEL, J.A., (1982), An application of the drawing block diagrams and thematic maps in the manipulation of spatial data bases, Comunicación al Urban Data Management 9th European Symposium, Valencia.
78. HORWOOD, E.M., (1970), "Grid co-ordinate geographic identification systems", en BISCO, R.L. (ed.), Data bases, computers and the social sciences, pp. 20-37.
79. HOWARTH, R.J., (1971), "Fortran IV program for grey-level mapping of spatial data", Jour. Intern. Assoc. Mathe. Geol., 3, pp. 95-121.
80. HUKÉ, R.E., FIELDING, G.J. y RUMAGE, K.W. (eds.), (1969), Computer assisted instruction in Geography, AAG Commission on College Geography, Technical paper 2.
81. JARO, M.A., (1972), GRIDS: A computer mapping system.
82. JEFFERY, M., O'HARE, H. y BOARD, C., (1974), Coropleth mapping on the microfilm plotter: an attempt to improve the graphic quality of automated maps, Comunicación presentada al 7th International Conference on Cartography, Madrid.
83. JENKS, G.F. y KNOS, D.S., (1961), "The use of shaded patterns in graded series", A.A.A.G., 51, pp. 316-34.
84. JENKS, G.F., (1981), "Lines, computers, and human frailties", A.A.A.G., vol 71, nº 1, pp. 1-10.
85. JONES, J.G.T., (1972), Diagrams drawn by computer.
86. KADMON, N., (1971), "KOMPLOTT: 'Do-it-yourself' computer cartography", Cartographic Journal, 8, pp. 139-144.



87. KADMON, N., (1974), Computer-aided vs. computer maps, Comunicación al 7th International Conference on Cartography, Madrid.
88. KAO, R.C., (1963), "The use of Computers in the Processing and Analysis of Geographic Information", Geographical Review, vol LIII, pp. 530-547.
89. KERN, R. y RUSHTON, G., (1969), "MAPIT: A computer program for production of flow maps, dot maps and graduated symbol maps", Cartographic Journal, 6, pp. 131-137.
90. KIRBY, A.M. y TARN, D., (1976), "Some problems of mapping in 1971 census by computers", Environment and Planning, 8.
91. KIRK, M.V. y PRESON, D.A., (1971), "Fortran IV programs for computation and printer display of maps and mathematically defined surfaces", Geocom Programs, 3.
92. KLINGER, A., (1977), Data structures, computer graphics and pattern recognition, New York, Academic Press.
93. KNOWLTON, R.C., (1965), "Computer-Produced Movies", Science, vol 150, pp. 116-120.
94. KOELLING, M.E.V. y WHITTEN, E.H.T., (1973), "Fortran IV program for spline surface interpolation and contour map production", Geocom Bulletin, 6.
95. KOELLING, M.E.V. y WHITTEN, E.H.T., (1973), "Spline-surface interpolation, spatial filtering and trend surfaces for geologic mapped variables", Journ. Intern. Assoc. Mathem. Geol., 5 pp. 111-126.
96. LEE, P.J., (1969), "Fortran IV programs for canonical correlation and canonical trend surface analysis", Computer Contributions, 32, State Geological Survey, Lawrence, Kansas.
97. MCDUGALL, E.B., (1976), Computer programming for Spatial Problems, London, Arnold, 160 pp.
98. MCINTYRE, D.B., POLLARD, D.D. y SMITH, R., (1968), "Computer programs for Automatic Contouring", Computer Contributions, 23, State Geological Survey, Lawrence, Kansas.
99. MACOMBER, L., (1971), "Utility of trend surfaces in inter-regional map comparisons", Journ. Regional Sci., 11, pp. 87-90.
100. MALING, D.H., (1973), Coordinate Systems and Map Projections, London, George Philip and Son, 255 pp.
101. MALLET, J.L., (1974), "Présentation d'un ensemble de méthodes et techniques de la cartographie numérique", Annales de l'Ecole Nationale Supérieure de Géologie Appliquée et Géochimiques, n°4, C.N.R.S., Nancy.

102. MARBLE, D.F., (1967), Some Computer Programs for Geographic Research, Evanston, Illinois, Department of Geography, Northwestern University.
103. MARINO, J.S., (1978), Characteristic Points and their significance in Cartographic Line Generalization, Masc. Dissertation, Univ. of Kansas.
104. MARTIN, F.F., (1968), Computer modelling and simulation.
105. MARTINEZ DE SOLA, I., (1981), Aproximación a la cartografía automática mediante ordenador. Análisis de sus posibilidades de uso en el marco de una investigación geográfica, Memoria de Licenciatura, Facultad de Geografía e Historia, U.C.M.
106. MATHER, P.M., (1977), Computers in Geography. A practical approach, Oxford, Basil Blackwell.
107. MERRIAM, D.F. y SNEATH, P.H.A., (1966), "Quantitative comparison of contour maps", Journ. Geophys. Research, 71, pp. 1105-1115.
108. MERRIAM, D. y COCKE (eds.), (1967), "Computer applications in the earth sciences: Colloquium on Trend Analysis", Computer Contributions, 12, State Geological Survey, Lawrence, Kansas.
109. MERRILL, R.D., (1973), Representation of contours and regions for efficient computer search, Communications of the A.C.M., 16.
110. MILLER, R.L., (1964), "Comparison analysis of trend surface maps" Stanford Univ. Publ. Geol. Sci., 9, pp. 669-692.
111. MOELLERING, H., (1972), Traffic Crashes in Washtenaw County Michigan. 1968-1970, Highway Safety Research Institute, Univ. of Michigan.
112. MOELLERING, H., (1973), "The Computer Animated Film: A Dynamic Cartography", Proceedings of Association for Computing Machinery, pp. 64-69.
113. MOELLERING, H., (1973), "The potential uses of a Computer Animated Film in the Analysis of Geographical Patterns of Traffic Crashes", Accident Analysis and Prevention, 8, pp. 215-227.
114. MOELLERING, H., (1977), Real and virtual maps, Comunicación al 1977 meeting of the Association of American Geographers, Salt Lake City.
115. MOELLERING, H., (1978), An approach to the Real Time Cartographic Animation of three dimensional objects, Comunicación presentada al Ninth International Conference in Cartography, Maryland.

116. MONMONIER, M.S., (1965), "The production of shaded maps on the digital computer", Professional Geographer, 17, pp. 13-14.
117. MONMONIER, M.S., (1969), "Computer mapping with the digital incremental plotter", Professional Geographer, 20, pp. 408-409.
118. MONMONIER, M.S., (1970), "Shaded area symbols for the digital incremental plotter", Tijdschrift voor Economische en Sociale Geografie, 61, pp. 374-378.
119. MONMONIER, M.S., (1978), "Viewing Azimut and map clarity", A.A.A.G., vol 68, n° 2, pp. 180-195.
120. MONMONIER, M.S., (1982), Computer-Assisted Cartography. Principles and Prospects, 214 pp.
121. MORENO JIMENEZ, A., (1979), Crecimiento y estructura suburbana en el SW de Madrid: Carabanchel, Tesis Doctoral, Facultad de Geografía e Historia, U.C.M.
122. MORENO JIMENEZ, A., (1979), "Pautas de localización intraurbana de la industria en el SW de Madrid (Carabanchel): Una aproximación metodológica", Estudios Geográficos, n° 156-157, pp. 435-461.
123. MORENO JIMENEZ, A., (1981), "Análisis de la calidad de la vivienda y perspectiva de la función residencial en el Centro", en Madrid: Estudios de Geografía Urbana, Instituto Juan Sebastián Elcano, C.S.I.C., Madrid, pp. 185-208.
124. MURRAY, R.D. (ed.), (1970), Computer handling of Geographical Information, Washington, Society of Photographic Scientists and Engineers.
125. NEWMAN, N.M. y SPROULL, R.F., (1973), Principles of interactive computer-graphics, New York, MacGraw-Hill.
126. NÚÑEZ DE LAS CUEVAS, R., (1970), "Estado actual de la automatización del proceso cartográfico", Ciudad y Territorio, n° 2.
127. OLEA, R.A., (1974), "Optimal contour mapping using universal kriging", Journal of Geophysics Research, 79, pp. 695-702.
128. OLSON, J.M., (1981), "Spectrally encoded two-variable maps", A.A.A.G., vol 71, n° 2, pp. 259-276.
129. PARSLow, R.D. y GREEN, R.E., (1971), Advanced computer-graphics.
130. PETIT, J., (1972), Cartographie automatique et aménagement du territoire, Rapport du Centre de traitement visuel de l'information et de la cartographie économique et sociale.
131. PEUCKER, T., (1972), Computer Cartography, Washington, Commission on College Geography, Association of American Geographers.

132. PEUCKER, T.K. y CHRISMAN, N., (1975), "Cartographic Data Structures", The American Cartographer, vol 2, n° 1.
133. REQUES VELASCO, P., (1982), Segovia: Emigración rural y crisis demográfica, Tesis Doctoral, Facultad de Geografía e Historia, U.C.M.
134. RHIND, D.W., (1971), "Automated contouring. An empirical evaluation of some differing techniques", Cartographic Journal, 8, pp. 145-158.
135. RHIND, D.W., (1974), "An introduction to the digitizing and edition of mapped data", en Automated Cartography, Special Publication number 1, London, British Cartographic Society, pp. 50-68.
136. RHIND, D.W., (1976), "Towards universal, intelligent and usable automated cartographic systems", ITC Journal, 4, pp 515-45.
137. RHIND, D.W., (1980), "A multi-temporal, multi-purpose, multi-user, multi-resolution geographical information system", U.R.P.I.S. 7, Proceedings of the Australian Urban and Regional Information Systems Association annual conference, Newcastle.
138. RHIND, D.W., (1981), "Geographical information systems in Britain", en BENNETT, R.J. y WRIGLEY, N. (eds.), Quantitative Geography in Britain: Retrospect and Prospect, London, Routledge and Kegan Paul, pp. 17-35.
139. RHODES, J.M., (1969), "The application of cluster and discriminatory analysis in mapping granite intrusions", Lithos, 2, pp. 223-237.
140. RIMBERT, S., (1980), "Aperçu général sur la cartographie expérimentale", Recherches Géographiques à Strasbourg, 8, pp. 3-24.
141. ROBINSON, A.H., (1962), "Mapping the correspondence of isarithmic maps", A.A.A.G., 52, pp. 414-425.
142. ROHLF, F.J., (1969), "GRAFPAC, graphical output subroutines for the GE 635 computer", Computer Contribution, 36.
143. ROJO PEREZ, F., (1981), Aproximación a la Geografía de la Infancia. Estudio socioespacial de los centros educativo-asistenciales de la primera edad. Las guarderías infantiles del distrito de San Blas, Memoria de Licenciatura, Facultad de Geografía e Historia, U.C.M.
144. ROJO PEREZ, F., (1982), "El ordenador como instrumento de trabajo aplicado a estudios de Geografía de la infancia. Entrada y manipulación de la información", Estudios Geográficos, C.S.I. C., (en prensa).

145. ROSING, K.E., (1969), "Computer graphics", Area, 3, pp. 2-7.
146. ROWLES, R., (1977), Perception of Perspective Block Diagrams, Comunicación presentada al 1977 meeting of the Association of American Geographers, Salt Lake City.
147. SAEZ CALA, A., (1982), El desempleo en el distrito madrileño de San Blas, Memoria de Licenciatura, Facultad de Geografía e Historia, U.C.M.
148. SAEZ CALA, A., VIDAL DOMINGUEZ, M.J. y ROJO PEREZ, F., (1982), Introducción al Programa de Cartografía Automática por impresora de líneas: SYMAP, Madrid, Editorial U.C.M., (en prensa).
149. SANTOS PRECIADO, J.M. y MUGURUZA CAÑAS, C., (1982), Introducción a la utilización del paquete de programas estadísticos BMDP en el análisis de datos geográficos, Madrid, Editorial U.C.M., (en prensa).
150. SAWYER, J.S., (1960), "Graphical output from computers and the production of numerically forecast or analysed synoptic charts", Meteorological Magazine, 89, pp. 187-190.
151. SEN GUPTA, A.K., (1974), Automated cartography system, Comunicación al 7th International Conference on Cartography, Madrid.
152. SENTANCE, W.A., (1969), "The graphic representation of computer output" Tijdschrift voor Economische en Sociale Geografie, 60, pp. 180-86.
153. SHEPHARD, D., (1968), "A two-dimensional interpolation function for irregularly spaced data", Proceedings Association of Computing Machinery, 11, pp. 517-523.
154. SHEPHARD, D., (1970), Computer Mapping as an aid in pollution studies, Individual report, Harvard University Press.
155. SHEPHARD, D., (1976), "Bridge that gap with GAPE: Geographical Algorithms and Program Exchange", Area, 8, 3, pp. 173-4.
156. SMITH, D., (1975), Patterns in Human Geography. An introduction to Numerical Methods, Brunel House Newton Abbot Devon, David and Charles, 373 pp.
157. SPRUNT, B.F., (1972), Contours in perspective, Paper presented at Institute of British Geographers Annual Conference, Aberdeen.
158. SPRUNT, B.F., (1973), Relief representation in automated cartography: an algorithmic approach, Comunicación al Nato Advanced Study Institute, Nottingham.
159. STARBUCK, W.H. y DUTTON, J.M., (1971), "The history of simulation models", en DUTTON, J.M. y STARBUCK, W.H. (eds.), Computer simulation of human behaviour, pp. 9-102.

160. TAYLOR, D.R. y FRASER (eds.), The Computer in Contemporary Cartography, Chichester, Wiley and Sons, 252 pp.
161. THAYER, R.P. y STORER, R.F., (1969), "Scale selection for computer plots". Applied Statist., 18, pp. 206-8.
162. THOMAS, A.N., (1976), Spatial models in computer-based information systems, Ph.D., Univ. of Edinburgh.
163. THROWER, N.J.W., (1959), "Animated Cartography", Professional Geographer, 11, pp. 9-12.
164. THROWER, N.J.W., (1961), "Animated Cartography in the United States" en International Yearbook of Cartography, pp. 20-29.
165. TOBLER, W.R., (1959), "Automation and Cartography", Geographical Review, 49, pp. 526-534.
166. TOBLER, W.R., (1962), "A Classification of Map Projections", A.A.A.G., vol 52, pp. 167-75.
167. TOBLER, W.R., (1965), "Automation in the preparation of thematic maps", Cartographic Journal, 2, pp. 32-38.
168. TOBLER, W.R., (1970), "A computer movie simulating urban growth in Detroit Region", Economic Geography, 46, pp. 234-40.
169. TOBLER, W.R. (ed.), (1970), Selected computer programs, Ann Arbor: Department of Geography, Univ. of Michigan.
170. TOMLINSON, R.F. (ed.), (1970), Environmental Information Systems, Ottawa, International Geographical Union Commission on Geographical Data Sensing and Processing.
171. TOMLINSON, R.F. (ed.), (1972), Geographical Data Handling, Ottawa, International Geographical Union Commission on Geographical Data Sensing and Processing.
172. TOMLINSON, R.F., (1974), The application of electronic computing methods to the storage, compilation and assement of mapped data, Ph.D., Univ. of London.
173. TRAYLOR, Ch.T., (1979), The evaluation of a Methodology to Measure Manual Digitization Error in Cartographic Data Bases, Ph.D., Univ. of Kansas.
174. TUERKE, K., (1975), "Computer aided digitizing of boundary networks", en Automation in Cartography, Technical Working Session, Enschede, I.C.A.
175. UTANO, J.J., (1980), "Computer programs 2: a program to draw four types of statistical maps in a vector plotter", Environment and Planning, vol 12, n°8.

176. UTTING, J. y HALL, J., (1973), The use of computers in University Social Science Departments.
177. VIRGINIA POLYTECHNIC INSTITUTE AND STATE UNIVERSITY, (1975), Computer Applications in Underground Mining Systems.
178. WAUGH, T.C., (1974), Geographic data in a system environment, Msc. Dissertation, Univ. of Edinburgh.
179. WAUGH, T.C., (1981), "The intelligent map", Canadian Cartographer.
180. WILSON, E., (1966), "The desing and selection of graded shadings for black and white maps", Bulletin of the Society of University Cartographers, pp. 11-16.

