

# UNA APLICACION INFORMATICA PARA EL ANALISIS GRANULOMETRICO: EL PROGRAMA DUNE

Eduardo Arana  
Itxaro Latasa Zaballos

---

*Datu-prozesuaren informatizazioak gaur artean denbora luzea zeramaten eginkizunak aintzea ekarri du. Analisi granulometrikoek eskatzen duten kalkulu kopuruak prozesua automatizatzeko programa informatiko bat egitea justifikatzen du. DUNE programak analisi granulometrikoan erabili ohi diren indizeen kalkulua burutzen du, lorturiko emaitzez osaturiko datu-base bat kudeatzen du eta fitxak zein zerrenden inprimaketa bideratzen du.*

*La incorporación de la informática al proceso de datos ha permitido agilizar tareas que hasta ahora suponían una gran inversión de tiempo. La cantidad de cálculos que requieren los análisis granulométricos justifica la realización de un programa informático que automatice el proceso. El programa DUNE permite el cálculo de los índices más utilizados en el análisis granulométrico, gestiona una base de datos con los resultados obtenidos y permite imprimir fichas y listados.*

*L'incorporation de l'informatique au traitement de l'information a permis d'accélérer des tâches auxquelles dans le passé il fallait consacrer beaucoup de temps. Le grand nombre de calculs que précisent les analyses granulométriques justifient la réalisation d'un programme d'informatique qui automatise tout le processus. Le programme DUNE permet de calculer les indices les plus utilisés dans l'analyse granulométrique, organise une banque de données avec les résultats obtenus et permet d'imprimer des fiches et des répertoires.*

## INTRODUCCION

Reconstruir los ambientes sedimentarios antiguos y estudiar los procesos morfogenéticos actuales puede hacer imprescindible la aplicación de técnicas sedimentológicas. Entre éstas el análisis granulométrico, por su sencillez, economía y óptimos resultados es frecuentemente utilizada.

Las diferentes fases que debemos realizar en este tipo de análisis van desde el tratamiento de las muestras en el laboratorio hasta la elaboración de gráficas e índices estadísticos. Puesto que estas fases se realizan casi siempre de forma manual, el análisis granulométrico acaba siendo, pese a su sencillez, un trabajo lento y pesado. Aunque no se pueda hacer mucho por facilitar el trabajo de laboratorio, sí podemos intentar agilizar el proceso posterior de la información. Por esto, hace tiempo que teníamos la intención de elaborar un programa que facilitase el análisis de los datos.

Cuando este programa era sólo una idea, Félix Ugarte nos animó a llevarlo a cabo, considerando que podía resultar de gran utilidad. Pensamos que esta publicación de homenaje a Félix Ugarte es el lugar adecuado para dar a conocer una primera versión del programa DUNE.

## EL PROGRAMA

Al intentar hacer un programa que permitiese realizar con facilidad los análisis granulométricos, pensamos que éste no podía limitarse a analizar los datos y presentar los resultados. Sería útil poder almacenar los datos resultantes para recuperarlos posteriormente, editarlos y borrarlos si es preciso. Todo este planteamiento nos llevó al terreno de las bases de datos y más concretamente a dBASE III, y su lenguaje de programación.

Tradicionalmente los lenguajes de programación utilizados para aplicaciones de ingeniería y matemáticas han sido Fortran, Pascal, C e incluso Basic. El lenguaje de programación dBASE está considerado como propio de las tareas de gestión, pero no apto para otros usos. Por nuestra parte pensamos que el lenguaje dBASE, y más concretamente la versión del compilador CLIPPER de Nantuckett, cumple los requisitos necesarios para ser utilizado en un programa que tiene como tarea principal la realización de cálculos matemáticos.

Nuestra intención inicial era realizar el programa mediante el lenguaje CLIPPER, pero esto nos habría impedido presentarlo en forma de listado. Por esto hemos optado por presentar una primera versión del programa DUNE escrita en el lenguaje dBASE.

El programa DUNE, en la versión de demostración que presentamos, además del análisis de los datos pone a disposición del usuario las siguientes opciones:

*Almacenar* una serie de *tamices*, pudiendo alterarla si es necesario. Esta serie se guarda en un archivo, evitando así tener que volver a escribirla cada vez que iniciamos una se-

si3n de trabajo. Una utilidad de base de datos para gestionar nuestros an3lisis sin necesidad de conocer el funcionamiento del programa dBASE. Posibilidad de *imprimir un listado* ordenado de toda o parte de nuestra base de datos. Una opci3n de *conversi3n de escalas* entre mil3metros y phi, que proporciona una precisi3n muy superior a la del m3todo gr3fico. Adem3s, podremos limpiar y *ordenar* nuestras bases de datos.

### INSTRUCCIONES DE USO

A continuaci3n exponemos los pasos que deberemos dar para disponer del programa DUNE. Lo primero que hay que hacer es escribir los archivos INICIO.PRG, DUNE.PRG y SEGUNDO.PRG que aparecen en el listado que presentamos. Estos archivos deben grabarse en el formato ASCII. Todos los procesadores de texto incorporan la posibilidad de generar archivos ASCII, bastar3a consultar el manual de nuestro procesador.

En segundo lugar deberemos crear las dos bases de datos que necesita el programa. (Si es necesario consultar el manual de dBASE para crear las bases de datos.) Para ello utilizaremos dBASE III y crearemos las bases de datos con la estructura siguiente:

BASE DE DATOS: BASE1 .DBF

CAMPO	TIPO	ANCH.	DECIMAL
C1	C	20	
C2	C	20	
C3	C	20	
C4	C	20	
C5	F	8	
C6	F	8	
C7	C	5	
C8	C	10	
C9	C	8	4
C10	N	8	4
C11	N	8	4
C12	N	8	4
C13	N	8	4
C14	N	9	5
C15	N	9	5
C16	N	9	5
C17	N	9	5
C18	N	9	5
C19	N	9	5
C20	N	9	5
C21	N	9	5
C22	N	9	5
C23	N	9	5
C24	N	9	5
C25	N	9	5
C26	N	9	5
C27	C	225	

BASE DE DATOS:BASE2.DBF

CAMPO	TIPO	ANCHURA	DECIMAL
MILIMETROS	N	7	4
MASACUM	N	8	4

Situaremos todos los ficheros en el subdirectorio en que tengamos el programa dBASE III y, una vez dentro del programa dBASE, ejecutaremos el programa INICIO pulsando en el punto indicador la orden DO INICIO. El programa inicio preparará las bases de datos que hemos creado para que puedan ser utilizadas por el programa DUNE y creará los archivos de índice necesarios. Sólo es necesario ejecutar el programa inicio cuando instalemos DUNE por primera vez.

El último paso es llamar al programa DUNE escribiendo DO DUNE en el punto indicador de dBASE. Veremos a continuación algunas indicaciones sobre el uso del programa.

## GESTION DE TAMICES

Desde el menú principal de DUNE elegiremos la opción GESTION DE TAMICES para introducir las luces de malla, en milímetros, de nuestra serie de tamices. La primera vez que utilicemos el programa no aparecerá ningún tamiz en pantalla, por lo que pasaremos a introducir los valores de la serie de tamices que queramos utilizar. Es importante recordar que siempre deberemos incorporar como primer tamiz el utilizado para separar los materiales tipo grava aunque no tengamos en cuenta los materiales retenidos en ese tamiz. Si no lo hacemos así el programa puede tener problemas para realizar los cálculos necesarios en muestras con abundancia de materiales gruesos.

Cuando introduzcamos el primer tamiz el programa nos preguntará si deseamos introducir más tamices. Pulsaremos S y seguiremos introduciendo nuestra serie de tamices. Al terminar pulsaremos en primer lugar la tecla <Ctrl> y manteniéndola pulsada pulsaremos tecla <END>. Esta serie de tamices queda almacenada y será la que utilice el programa durante todas las sesiones de trabajo hasta que realicemos algún cambio.

Podemos cambiar el valor de cualquier tamiz, añadir tamices o borrarlos. Para borrar un tamiz posicionaremos el cursor en el valor que queramos borrar, pulsaremos la tecla <Ctrl> y manteniéndola pulsada pulsaremos la tecla <U>. El tamiz no desaparecerá de la pantalla pero en la esquina inferior derecha de la línea de información observaremos que aparece "Del" indicando que el tamiz ha sido borrado. La próxima vez que entremos en GESTION DE TAMICES el tamiz borrado no aparecerá.

Para añadir un nuevo tamiz basta con posicionar el cursor en el último tamiz de la serie e intentar desplazarlo hacia abajo. El programa preguntará si deseamos añadir más registros y pulsando S podremos incluir nuevos tamices. El nuevo tamiz se añade siempre al final de la lista y no en su posición dentro de la lista ordenada de tamices; tras cualquier cambio el programa ordenará automáticamente la serie de tamices de mayor a menor en cuanto movamos el cursor.

## ANALISIS DE DATOS

Tras almacenar nuestra serie de tamices pasaremos a realizar el análisis de los datos eligiendo la opción ANALISIS DE DATOS en el menú principal. Pasaremos a una segunda pan-

talla en la que podremos elegir entre INTRODUCCION DE PERCENTILES MANUAL y CALCULO AUTOMATICO.

DUNE puede realizar los análisis partiendo de dos tipos de datos: los percentiles ya calculados en escala phi o las masas acumuladas para cada tamiz. Es posible que algún usuario prefiera calcular él mismo los percentiles por el método gráfico para realizar sus análisis, pero no es esta la razón para incluir esta opción en el programa puesto que el sistema de cálculo de los percentiles de DUNE es más preciso que cualquier método gráfico. Esta opción se incorpora para permitirnos comprobar los análisis de otros estudios cuando éstos nos proporcionan los valores de los percentiles para sus muestras y no las masas por tamiz.

En la opción de análisis automático sólo es necesario introducir en el programa la masa acumulada para cada uno de los tamices de la serie. (Hay que recordar que en el tamiz utilizado para separar las gravas debemos poner siempre como masa el valor cero, puesto que ese material es separado de la fracción arenosa.) La pantalla que presenta DUNE es idéntica a la de GESTION DE TAMICES salvo que ahora no podremos editar ni añadir tamices.

Una vez introducidos los datos de masa acumulada para cada tamiz pulsaremos <Ctrl>+<End> y DUNE calculará los percentiles y los índices necesarios, presentando una ficha en la que añadiremos el resto de datos de identificación de la muestra: nombre, lugar de toma, trabajo al que pertenece, etc. Cualquiera de estos campos puede dejarse en blanco, aunque si no damos nombre a la muestra analizada no podremos recuperarla con la utilidad de base de datos.

Tras rellenar estos campos aparecerá una línea en la base de la pantalla que nos ofrecerá dos posibilidades: imprimir la ficha o regresar al menú principal.

## **BASE DE DATOS**

DUNE incorpora una opción de BASE DE DATOS que permite al usuario localizar un análisis de la base de datos por su nombre, lugar de recogida de la muestra o por el trabajo para el que se realizó. Para facilitar la búsqueda de un determinado análisis o grupo de análisis podemos localizar cualquier muestra con solo introducir una parte cualquiera del dato a buscar: nombre, lugar o trabajo. El programa localizará una a una todas las muestras que cumplan las condiciones que solicitemos y nos permitirá seguir la búsqueda por la base de datos, editar una muestra, borrarla, o imprimir una ficha con todos los datos.

## **LISTADOS**

Puesto que en algunas ocasiones desearemos tener nuestros análisis impresos en papel, DUNE incorpora una opción de listados en la que podemos elegir entre listar todos los análisis de la base de datos o únicamente los que corresponden a un trabajo determinado. Estos listados pueden ordenarse por nombre, por lugar y, en el caso de listar toda la base de datos, por trabajo. Los listados de DUNE se realizan en letra comprimida, que el programa selecciona automáticamente, lo que permite imprimir un listado de 132 columnas en una impresora de carro estrecho.

## **CONVERSION PHI <-> MM**

La opción de conversión entre las escalas phi y milímetros nos permite pasar cualquier dato de una escala a otra, siempre que el valor en milímetros no sea cero.

## ORDENAR BASE DE DATOS

La última opción del menú principal de DUNE nos permite limpiar y ordenar nuestras bases de datos cuando hayamos borrado muchos registros y queramos eliminarlos definitivamente, aumentando así la velocidad de funcionamiento del programa. Cuando la opción BORRAR REGISTROS de la utilidad de base de datos elimina un registro, éste permanece en la base de datos, ocupando una cierta cantidad de espacio en disco. Esto no es un problema si hemos borrado pocos registros pero puede llegar a entretener el funcionamiento del programa cuando hay muchos registros borrados.

También hay que tener en cuenta que los registros se almacenan en el orden en que los introducimos, orden que no es siempre el más adecuado para las tareas de búsqueda. Si reordenamos la base de datos cada cierto tiempo el programa DUNE trabajará con una serie ordenada de registros, lo que acelerará las labores de búsqueda. Por esto DUNE ofrece una opción ORDENAR BASE DATOS que elimina el espacio ocupado por los registros borrados y ordena las bases de datos.

## INDICES UTILIZADOS POR DUNE

Para calcular los índices son necesarios los valores en phi de los percentiles acumulados 5, 16, 25, 50, 75, 84 y 95. Si no es posible hallar todos estos valores partiendo de los datos de masa acumulada por tamiz, el programa puede bloquearse. Siempre que recordemos incluir en la lista de tamices el utilizado para separar las gravas, DUNE no tendrá problemas para calcular los percentiles inferiores, Para evitar problemas en el cálculo de los percentiles superiores debemos asegurarnos de que el último tamiz de la serie que utilicemos contiene al menos el 95% de la masa acumulada. Si no es así, el programa no podrá calcular el percentil 95, bloqueándose.

A continuación indicamos cuáles son los índices que calcula DUNE y las fórmulas utilizadas. Los valores p5, p16, etc representan los percentiles acumulados en escala phi. Los valores m25, m50, etc representan los percentiles acumulados en mm.

La *mediana* (percentil 50) se indica en mm. Al representar el diámetro de grano que divide la muestra en dos partes iguales, la escala en mm es más fácilmente visualizable que la phi. Si es necesario disponer de este índice en escala phi podemos recurrir a la utilidad de conversión de escalas de DUNE.

El índice Mz de Folk y Ward (1957), utilizado como aproximación a la media, se calcula en escala phi con la fórmula siguiente:

$$Mz = \frac{p16 + p50 + p84}{3}$$

El índice de clasificación So (Sorting) de Trask (1932) presenta ciertos problemas puesto que hay desacuerdos en la bibliografía consultada sobre si debe realizarse en escala phi o en mm, e incluso sobre si la fórmula exacta es  $So = (m25/m75)^{1/2}$  o  $So = (m75/m25)^{1/2}$ . Respecto a la escala en que debe calcularse So, es fácil comprobar que si utilizamos la escala phi cabe la posibilidad de que el percentil 25 sea un número negativo y el percentil 50 uno positivo. Como consecuencia estaríamos intentando calcular una raíz cuadrada de un número negativo, algo que, no siendo imposible, no tiene sentido en este caso. No podemos resolver este problema convirtiendo el valor negativo a positivo, puesto que así incrementamos el valor del percentil 25 mejorando, falsamente, la clasificación de nuestra muestra.

Respecto al problema de la fórmula exacta de  $S_o$ , al no haber podido acceder al artículo original de Trask, hemos decidido calcular este índice partiendo de otro relacionado con él. Hemos utilizado el índice  $Q_d$  (Quartile deviation) de Krumbain (1936b), puesto que  $S_o$  es el antilogaritmo en base 2 de éste índice. La fórmula de  $Q_d$  y la utilizada por DUNE para calcular  $S_o$  son, en  $\phi$  y  $mm$  respectivamente, las siguientes:

$$Q_d = \frac{p_{75} - p_{25}}{2}$$

$$S_o = e^{(p_{75}-p_{25}) \times \ln 2 / 2}$$

El índice de clasificación  $\delta$  (sigma) de Folk y Ward (1957) se calcula en escala  $\phi$  con la fórmula siguiente:

$$\delta = \frac{p_{84} - p_{16} \quad p_{95} - p_5}{4 \quad 6.6}$$

El índice de asimetría (Skewness)  $S_k$  de Trask (1932) se calcula en  $mm$  con la fórmula siguiente:

$$S_k = \frac{m_{25} \times m_{75}}{m_5 \cdot 0^2}$$

La asimetría gráfica inclusiva  $S_{ki}$  (Inclusive Graphic Skewness) de Folk y Ward (1957) se calcula en escala  $\phi$  con la fórmula siguiente:

$$S_{ki} = \frac{p_{84} - p_{16} + 2p_{50} \quad p_{95} - p_5 - 2p_{50}}{2(p_{84} - p_{16}) \quad 2(p_{95} - p_5)}$$

El índice de apuntamiento  $K_g$  (Graphic Kurtosis) de Folk y Ward (1957) se calcula con la fórmula siguiente:

$$K_g = \frac{p_{95} - p_5}{2.44(p_{75} - p_{25})}$$

## ALGUNOS COMENTARIOS SOBRE EL PROGRAMA

Como ya hemos indicado, esta versión de demostración del programa DUNE está preparada para su difusión en forma de listado. Por esto lo hemos aligerado de todo lo que no fuera imprescindible, a fin de que el listado no resultara excesivamente largo e incómodo de escribir.

Una de las limitaciones es que DUNE accede a las bases de datos abriendo estas directamente en lugar de utilizar variables intermediarias. Como resultado el programa es más sensible a los cortes de tensión en la red, pero resulta más breve.

Tampoco ha sido posible incorporar una rutina de control de errores, aunque estos no tienen por qué aparecer si seguimos los pasos dados y tecleamos el programa correctamente.



te. Pese a todo, es probable que al ejecutar el programa por primera vez nos aparezcan mensajes de error debido a que hayamos escrito un punto en lugar de una coma o un paréntesis de menos. Lo que debemos hacer es tomar nota de la línea entera que el programa indica como errónea, del módulo y del programa en que está el error, abrir el programa con nuestro editor de texto, localizar el error y lo corregirlo. Siguiendo estos pasos los errores de escritura desaparecerán.

DUNE almacena en su base de datos los valores en phi de los percentiles, pero no los presenta en pantalla ni en listados ya que era necesario reducir todo lo posible las rutinas que se encargan de los listados. Guardamos los percentiles en la base de datos en previsión de que versiones posteriores del programa (en preparación) nos permitan imprimirlos.

Por último, DUNE almacena con cada muestra la información sobre los tamices utilizados y las masas acumuladas para cada uno de ellos. Aunque esta información no se utiliza en el programa, se almacena para que posteriores versiones de DUNE puedan recuperarla para realizar gráficas en pantalla. Estos datos se almacenan como una cadena de caracteres en el campo C27 de la base de datos BASE1.DBF. Puesto que este campo tiene una longitud de 225 caracteres y la información de cada tamiz con su masa acumulada ocupa 15 caracteres, podemos utilizar hasta 15 tamices. Si nuestra serie tiene más tamices se perderá la información de los tamices por encima de ese número.

## BIBLIOGRAFIA

- FOLK, R. L. (1966): "A review of grain-size parameters" *Sedimentology*, 6, pp. 73-93
- FOLK, R. L. y WARD, W. C. (1957): "Brazos River bar: a study in the significance of grain size parameters." *Jour. Sed. Petr.* Vol 27, pp. 3-26
- KRUMBEIN, W. C. (1936): "The use of quartile measures in describing and comparing sediments" *Am. J. Sci.*, 32:98-111
- RIVIERE, A. (1977): *Méthodes granulométriques. Techniques et interprétations.* Masson, Paris.
- TRASK, P. D. (1932): "Origin and environment of source sediments of Petroleum." Gulf. Publ. Co., Houston

```

#####
# programa inicio.prg
# Prepara las bases de datos del programa
# DUNE. Llamario con DO INICIO antes de
# trabajar con DUNE
#####
set talk off
set status off
if .not. file("base1.dbf")
clea
@ 10,10 say "FALTA BASE DE DATOS PRINCIPAL. DEBE CREARSE"
wait "Pulse una tecla"
retu
endi
if .not. file("base2.dbf")
clea
@ 10,10 say "FALTA BASE DE TAJICES. DEBE CREARSE"
wait "Pulse una tecla"
retu
endi
clea
@ 1,0 to 22,79 double
set colo to w+/n
@ 10,17 say "PREPARANDO BASES DE DATOS PARA SU USO CON DUNE"
set colo to
use base1
appe blank
inde on c1 to i11
inde on c2+c1 to i12
inde on c3+c1 to i13
inde on c2 to i14
use
use base2
appe blank
inde on -milímetros to i2
use
rename base1.dbf to st.dat.dun
rename base2.dbf to nd.dat.dun
clea
retu
##### fin del programa inicio.prg
#####
# Programa dune.prg versión dBASE 6/92
# Programa principal
#####
set date italian
set echo off
set bell off
set talk off
set conf off
set dele on
set scor off
set esca off
set safe off
set proc to segundo
publ in2,in3,in4,in5,in6,in7,id1,id2,id3,id4,id5,id6,;
id7,id8,id9,id10,id11,id12,id13,menu,;
valor,c1,c2,n_c,cadena
stor 0 to in2,in3,in4,in5,in6,in7,id1,id2,id3,id4,id5,;
id6,id7,id8,id9,id10,id11,id12,id13

```

```

c1="w/b,w+/r,b"
c2="w/b,w+/b,b"
n_c="w/n,w+/n,n"
cadena=""
if isco()
set colo to kcl
endif
do while .t.
clear
menu=0
@ 1,0 to 22,79 double
@ 3,24 to 6,55
@ 4,25 say repl(CHR(177),30)
@ 4,32 say "— D U N E ——"
@ 5,25 say repl(CHR(177),30)
@ 5,31 say "Versión dBASE III+"
@ 7,28 say "1. GESTION TAJICES"
@ 8,28 say "2. BASE DE DATOS"
@ 9,28 say "3. ANALISIS DATOS"
@ 10,28 say "4. LISTADOS"
@ 11,28 say "5. CONVERSION DE ESCALAS"
@ 12,28 say "6. ORDENAR BASE DATOS"
@ 13,28 say "0. SALIR"
@ 16,28 say "OPCION: " get menu pict "9" rang 0,6
read
do case
case menu=0
exit
case menu=1
do mod1
case menu=2
do mod2
case menu=3
do mod3
case menu=4
do mod4
case menu=5
do mod5
case menu=6
do mod6
endc
endd
quit
#####fin del programa DUNE.PRG
#####
# Programa segundo.prg. Es el archivo de
# procedimientos del programa DUNE.PRG
#####
proc mod1
clea
use nd.dat.dun index i2
brow fields milímetros nofollow nomenu
use
retu
#####
proc mod2
priv op
do while .t.
op=0
clear

```

```

@ 4,31 say "LOCALIZAR ANALISIS"
@ 6,32 say "1. POR NOMBRE"
@ 7,32 say "2. POR TRABAJO"
@ 8,32 say "3. POR LUGAR"
@ 9,32 say "0. Salir"
@ 1,0 to 22,79 double
@ 3,29 to 5, 50
@ 11,32 say "OPCION: " get op pict "9" rang 0,3
read
do case
case op=0
exit
case op=1
do mod9 with "NOMBRE :", "c1"
case op=2
do mod9 with "TRABAJO:", "c2"
case op=3
do mod9 with "LUGAR :", "c3"
endc
endd
retu
#####
proc mod3
priv q5,q16,q25,q50,q75,q84,q90,q95,op,mt,m25,m75,m50
stor 0 to q5,q16,q25,q50,q75,q84,q90,q95,op,mt,;
m25,m75,m50
clea
@ 1,0 to 22,79 double
@ 3,29 to 5,49
@ 4,31 say "ANALISIS DE DATOS"
@ 6,29 say "1. INTRODUCCION DE PERCENTILES MANUAL"
@ 7,29 say "2. CALCULO AUTOMATICO"
@ 8,29 say "0. Salir"
@ 10,29 say "OPCION"
@ 10,37 get op pict "9" rang 0,2
read
do case
case op=0
clea
retu
case op=1
@ 6,29 clear to 10,65
@ 6,24 to 18,55
@ 7,26 say "INTRODUCIR VALORES INDICADOS"
@ 8,33 say "(En escala phi)"
set conf on
@ 10,33 say "Q5..." get q5 pict "999.9999"
@ 11,33 say "Q16..." get q16 pict "999.9999"
@ 12,33 say "Q25..." get q25 pict "999.9999"
@ 13,33 say "Q50..." get q50 pict "999.9999"
@ 14,33 say "Q75..." get q75 pict "999.9999"
@ 15,33 say "Q84..." get q84 pict "999.9999"
@ 16,33 say "Q95..." get q95 pict "999.9999"
read
set conf off
case op=2
do mod8
clea
@ 10,10 say "Procesando los datos. Espere un momento"
go bott

at=masacum
go top
do mod11 with 5
q5=valor
do mod11 with 16
q16=valor
do mod11 with 25
q25=valor
do mod11 with 50
q50=valor
do mod11 with 75
q75=valor
do mod11 with 84
q84=valor
do mod11 with 95
q95=valor
endc
in3=(q84-q16)/4+(q95-q5)/6.6
in6=(q16+q84-2*q50)/(2*(q84-q16))+
(q5+q95-2*q50)/(2*(q95-q5))
in7=(q95-q5)/(2.44*(q75-q25))
in2=(q16+q50+q84)/3
in4=exp((q75-q25)/28*log(2))
m25=exp(-11*q25*log(2))
m75=exp(-11*q75*log(2))
m50=exp(-11*q50*log(2))
in5=m25*m75/m50^2
go top
do while .not. eof()
cadena=cadena+str(milímetros,7,4)+str(masacum,8,4)
skip
enddo
use st_dat.dun index i11,i12,i13,i14
appe blank
repl c21 with q5
repl c22 with q16
repl c23 with q25
repl c24 with q75
repl c25 with q84
repl c26 with q95
repl c14 with m50
repl c15 with in2
repl c16 with in3
repl c17 with in4
repl c18 with in5
repl c19 with in6
repl c20 with in7
repl c27 with cadena
cadena=""
do mod7
@ 1,33 say " COMPLETAR DATOS "
read
@ 1,1 to 22,79 double
@ 23,29 say "Retorno / Imprimir"
op=" "
do while .not. (op="RI")
op=uppe(chr(inke()))
endd
if op="I"
do mod10

```

```

endi
use
retzu
#####
proc mod4
clea
priv me,li,pa,lat,sup,inf,i,c,fi
stor 8 to li
stor 1 to pa
stor chr(186) to lat
stor chr(209)+repl(chr(205),7) to sup
stor chr(207)+repl(chr(205),7) to inf
stor chr(179) to i
stor repl(chr(205),20) to c
do while .t.
me=0
fi=space(20)
@ 0,1 to 22,79 double
@ 3,32 to 5,45
@ 4,35 say "LISTADOS"
@ 7,30 say "1. LISTAR TODA LA BASE DE DATOS"
@ 8,30 say "2. LISTAR DATOS DE UN TRABAJO"
@ 9,30 say "0. Salir"
@ 12,30 say "OPCION" get me picture "9" range 0,2
read
do case
case me=0
retu
case me=1
me=0
clea
@ 0,1 to 22,79 double
@ 3,32 to 5,45
@ 4,35 say "LISTADOS"
@ 7,30 say "1. ORDENADOS POR NOMBRE"
@ 8,30 say "2. ORDENADOS POR TRABAJO"
@ 9,30 say "3. ORDENADOS POR LUGAR"
@ 10,30 say "0. Salir"
@ 12,30 say "OPCION" get me pict "9" rang 0,3
case me=2
clea
@ 0,1 to 22,79 double
@ 3,32 to 5,45
@ 4,35 say "LISTADOS"
use st_dat.dun index i14
set exac on
do while .t.
@ 7,20 say "¿TRABAJO A LISTAR?: " get fi pict "@!"
read
if fi=space(20)
retu
else
seek fi
if .not. foun()
@ 23,15
wait "NO EXISTE. PULSE UNA TECLA E INDIQUE OTRO"
@ 23,0 clea
fi=space(20)
loop
else

```

```

use
exit
endi
endi
endi
set exac off
me=0
clea
@ 0,1 to 22,79 double
@ 3,32 to 5,45
@ 4,35 say "LISTADOS"
@ 7,30 say "1. ORDENADOS POR NOMBRE"
@ 8,30 say "2. ORDENADOS POR LUGAR"
@ 10,30 say "0. Salir"
@ 12,30 say "OPCION" get me pict "9" rang 0,2
endc
read
do case
case me=0
retu
case me=1
use st_dat.dun index i11
exit
case me=2
use st_dat.dun index i12
exit
case me=3
use st_dat.dun index i13
exit
endc
endd
if fi(<)space(20)
set filt to c2-trim(trim(fi))
go top
endi
clea
@ 10,10
wait "PREPARE IMPRESORA. PULSE UNA TECLA PARA SEGUIR"
clea
@ 10,10 say "IMPRIMIENDO"
set devi to print
@ prow(),pcol() say chr(15)
do while .not. eof()
if li=8
@ 5,22 say "PROGRAMA DUNE. LISTADO DE ANALISIS"
@ 6,0 say chr(201)+c+chr(209)+c+chr(209)+c+;
repl(sup,7)+chr(187)
@ 7,0 say lat+-----NOMBRE-----+i+;
+-----LUGAR-----+i+-----TRABAJO-----+i+;
i+"MEDIANA"+i+"Hz....."+i+chr(229)+*.....+;
i*"So. (mm)+i*"Sk. (mm)+i*"Ski....."+i+"Kg....."+lat
endi
@ li,0 say lat+c1+i+c2+i+c3+i+str(c14,7,4)+i+;
str(c15,7,4)+i+str(c16,7,4)+i+str(c17,7,4)+i+;
str(c18,7,4)+i+str(c19,7,4)+i+str(c20,7,4)+lat
stor li+1 to li
if li>62
@ li,0 say chr(200)+c+chr(207)+c+chr(207)+c+;
repl(inf,7)+chr(186)
@ li+1,30 say "pagina" *trim(istr(pa))

```

```

ejec
stor pa+1 to pa
stor 8 to li
endi
skip
endd
set filt to
use
if li>8
@ li,0 say chr(200)+c+chr(207)+c+chr(207)+c+;
repl(inf,7)+chr(188)
@ 64,30 say "pagina"+trim(str(pa))
ejec
endi
@ prow(),pcoll() say chr(18)
set devi to screen
retu
#####
proc mod5
priv mm,phi,menu
do while .t.
stor 0 to menu
clea
@ 1,0 to 22,79 double
@ 3,29 to 5,50
@ 4,31 say "CONVERSION ESCALAS"
@ 7,31 say "1. Phi -> mm"
@ 8,31 say "2. mm -> Phi"
@ 9,31 say "0. Salir"
@ 11,31 say "OPCION: " get menu pict "9" rang 0,2
read
if menu=0
exit
endi
do case
case menu=1
store 0 to mm,phi
@ 14,41 say "-> mm " get mm pict "02 99.9999"
clea gets
@ 14,28 say "Phi " get phi pict "99.9999"
read
mm=exp(-1#phi#log(2))
@ 14,41 say "-> mm " get mm pict "99.9999"
clea gets
@ 23,0
wait "Pulse una tecla para seguir."
case menu=2
do while .t.
store 0 to mm,phi
@ 14,41 say "-> Phi " get phi pict "02 99.9999"
clea gets
@ 14,29 say "mm " get mm pict "99.9999"
read
if mm<>0
exit
endi
endd
phi=-1#log(mm)/log(2)
@ 14,41 say "-> Phi " get phi pict "99.9999"
clea gets

```

```

@ 23,0
wait "Pulse una tecla para seguir."
endc
endd
retu
#####
proc mod6
clea
@ 10,20 say "ORDENANDO BASES DE DATOS. ESPEI"
rena st_dat.dun to auxi.dun
use auxi.dun index i11
copy to st_dat.dun
use st_dat.dun index i11,i12,i13,i14
rein
erase auxi.dun
rena nd_dat.dun to auxi.dun
use auxi.dun index i2
copy to nd_dat.dun
use nd_dat.dun index i2
rein
erase auxi.dun
use
retu
#####
proc mod7
clea
@ 1,1 to 22,79 double
@ 2,2 to 8,45
@ 2,48 to 10,75
@ 11,2 to 15,37
@ 11,39 to 15,69
@ 16,2 to 20,37
@ 16,39 to 20,69
@ 3,3 say "TRABAJO....."
@ 3,51 say "CARBONATOS."
@ 4,3 say "NOMBRE....."
@ 4,51 say "MUNSELL....."
@ 5,3 say "LUGAR....."
@ 5,50 say "% ARENA GRUESA...."
@ 6,3 say "TIPO CURVA.."
@ 6,50 say "% ARENA MEDIA...."
@ 7,3 say "F. TOMA....."
@ 7,25 say "F. ANALISIS.."
@ 7,50 say "% ARENA FINA....."
@ 8,50 say "% TOTAL ARENA...."
@ 9,50 say "% ARCILLAS-LINDOS."
@ 12,5 say "TENDENCIA CENTRAL"
@ 12,42 say "CLASIFICACION"
@ 17,5 say "ASIMETRIA (SKEWNESS)"
@ 17,42 say "APUNTAMIENTO (KURTOSIS)"
if isco()
set colo to kc2
else
set colo to kn_c
endi
@ 13,7 say "Mediana.." get c14
@ 14,7 say "Mz....." get c15
@ 13,40 say "So (mm)." get c17
@ 14,44 say chr(229)+"..." get c16
@ 18,7 say "Sk..(mm)." get c18

```

```

@ 19,7 say "Ski....." get c19
@ 18,44 say "Kg.." get c20
clea gets
if isco()
set colo to kcl
else
set colo to
endi
@ 3,15 get c2 pict "e!"
@ 4,15 get c1 pict "e!"
@ 5,15 get c3 pict "e!"
@ 6,15 get c4 pict "e!"
@ 7,15 get c5
@ 7,37 get c6
@ 3,62 get c7 pict "e!"
@ 4, 62 get c8 pict "e!"
@ 5,67 get c9 pict "999.9999" rang 0,100
@ 6,67 get c10 pict "999.9999" rang 0,100
@ 7,67 get c11 pict "999.9999" rang 0,100
@ 8,67 get c12 pict "999.9999" rang 0,100
@ 9,67 get c13 pict "999.9999" rang 0,100
retu
#####
proc mod8
clear
use nd_dat.dun index i2
do while .not. eof()
repl masacum with 0
skip
endd
go top
brow freeze masacum nomeno noappend
clea
retu
#####
proc mod9
para k,p
priv c,f,b
f=0
use st_dat.dun index i11, i12, i13, i14
set orde to 0
do while .t.
go top
c:=spac(20)
clea
@ 1,1 to 22,79 double
@ 11,19 to 13,60
@ 14,29 say "(En blanco para salir)"
@ 12,22 say k get c pict "e!"
read
if c:=spac(20)
clea
exit
endi
loca for tria(trim(c))%p
if .not. foun()
? chr(7)
clea
@ 10,20
wait "No existe ese registro. Pulse una tecla.

```

```

exit
endi
do while .t.
do mod7
@ 23,11 say ;
"Seguir /Borrar /Editar /Imprimir /Retorno"
if f=0
clea gets
else
read
f=0
endi
b=" "
do while .not. (b@"SREIB")
b:=uppe(chr(inke()))
endd
do case
case b=chr(83)
cont
if .not. foun()
? chr(7)
clea
@ 10,20
wait "No hay más registros. Pulse una tecla."
exit
endi
case b=chr(66)
dele
? chr(7)
clea
@ 10,20
wait "Registro borrado. Pulse una tecla "
exit
case b=chr(69)
f=1
case b=chr(73)
do mod10
case b=chr(82)
exit
endc
endd
endd
use
retu
#####
proc mod10
priv a1,a2,a3,p1,p2,p3,p4
a1:=spac(1)
a2:=spac(2)
a3:=spac(3)
p1=chr(186)+a1+chr(179)
p2=chr(179)+a2+chr(179)
p3=chr(179)+a3+chr(186)
p4=chr(179)+a1+chr(179)
set devi to print
@ 1,0 say chr(201)+repl(chr(205),78)+chr(187)
@ 2,0 say chr(186)+a1+chr(218)+repl(chr(196),42)+;
chr(191)+a2+chr(218)+repl(chr(196),26)+chr(191)+;
a3+chr(186)
@ 3,0 say p1+"TRABAJO....."+c2

```

```

@ 3,45 say p2+* CARBONATOS.*+c7+spac(8)+p3
@ 4,0 say p1+*NOMBRE.....*c1
@ 4,45 say p2+* MUNSELL.....*+c8+a3+p3
@ 5,0 say p1+*LUGAR.....*c3
@ 5,45 say p2+* % ARENA GRUESA..."+str(c9,8,4)+p3
@ 6,0 say p1+*TIPO CURVA..."+c4
@ 6,45 say p2+* % ARENA MEDIA..."+str(c10,8,4)+p3
@ 7,0 say p1+*F. TONA....."+dtoc(c5)
@ 7,25 say "F.ANALISIS..."+dtoc(c6)+p2
@ 7,50 say "% ARENA FINA..."+str(c11,8,4)+p3
@ 8,0 say chr(186)+a1+chr(192)+repl(chr(196),42)+;
chr(217)
@ 8,48 say chr(179)+* % TOTAL ARENA...."+;
str(c12,8,4)+p3
@ 9,0 say chr(186)
@ 9,48 say chr(179)+* % ARCILLAS-LIMOS.*"+;
str(c13,8,4)+p3
@ 10,0 say chr(186)
@ 10,48 say chr(192)+repl(chr(196),26)+chr(217)+;
a3+chr(186)
@ 11,0 say chr(186)+a1+chr(218)+repl(chr(196),34)+;
chr(191)+a1+chr(218)+repl(chr(196),29)+;
chr(191)+spac(9)+chr(186)
@ 12,0 say p1+a2+*TENDENCIA CENTRAL"
@ 12,37 say p4+a2+*CLASIFICACION"
@ 12,69 say chr(179)+spac(9)+chr(186)
@ 13,0 say p1+a1+*Mediana.(mm)*+str(c14,9,5)
@ 13,37 say p4+*So (mm)."+str(c17,9,5)
@ 13,69 say chr(179)+spac(9)+chr(186)
@ 14,0 say p1+a2+a2+*Mz....."+str(c15,9,5)
@ 14,37 say p4+a2+a2+chr(229)+*..."+str(c16,9,5)
@ 14,69 say chr(179)+spac(9)+chr(186)
@ 15,0 say chr(186)+a1+chr(192)+repl(chr(196),34)+chr(217)+;
a1+chr(192)+repl(chr(196),29)+chr(217)+spac(9)+chr(186)
@ 16,0 say chr(186)+a1+chr(218)+repl(chr(196),34)+chr(191)+;
a1+chr(218)+repl(chr(196),29)+chr(191)+spac(9)+chr(186)
@ 17,0 say p1+a2+*ASINETRIA (SKEWNESS)"
@ 17,37 say p4+a2+*APLANTAMIENTO (KURTOSIS)"
@ 17,69 say chr(179)+spac(9)+chr(186)

@ 18,0 say p1+a2+a2+*Sk (mm)."+str(c18,9,5)
@ 18,37 say p4+a2+a2+*Kg.."+str(c20,9,5)
@ 18,69 say chr(179)+spac(9)+chr(186)
@ 19,0 say p1+a2+a2+*Ski....."+str(c19,9,5)
@ 19,37 say p4+spac(29)+chr(179)+spac(9)+chr(186)
@ 20,0 say chr(186)+a1+chr(192)+repl(chr(196),34)+chr(217)+;
a1+chr(192)+repl(chr(196),29)+chr(217)+spac(9)+chr(186)
@ 21,0 say chr(186)+spac(78)+chr(186)
@ 22,0 say chr(200)+repl(chr(205),78)+chr(188)
ejec
set devi to screen
retu
*****
proc mod11
param q
priv tam_min,tam_max,hallado,m_min,m_max,val_q
stor 0 to tam_min,tam_max,hallado,m_min,m_max
val_q=q*at/100
do while hallado=0
if val_q=masacum
stor -1*log(milimetros)/log(2) to valor
hallado=1
endi
if val_q>masacum
stor -1*log(milimetros)/log(2) to tam_min
stor masacum to m_min
skip 1
endi
if val_q<masacum
stor -1*log(milimetros)/log(2) to tam_max
stor masacum to m_max
hallado=1
skip -1
endi
endd
valor=tam_min+(tam_max-tam_min)/(m_max-m_min)*(val_q-m_min)
retu
*****fin del programa segundo.prg

```