

TÉCNICAS PEDAGÓGICAS Y ESTUDIO DE RESULTADOS PARA LA ENSEÑANZA DE MS-DOS Y LENGUAJE ENSAMBLADOR EN PRIMER CURSO DE I.T. INFORMÁTICA GESTIÓN / SISTEMAS.

Antonio J. de Vicente¹, José Gallego², Juan I. Pérez³,

Agustín Martínez⁴, Sara García⁵

Departamento de Automática. Universidad de Alcalá

¹*e-mail: avicente@aut.alcala.es*

²*e-mail: jgallego@aut.alcala.es*

³*e-mail: nacho@aut.alcala.es*

⁴*e-mail: hellin@aut.alcala.es*

⁵*e-mail: sara@aut.alcala.es*

RESUMEN: En el presente artículo se recoge la experiencia docente en la enseñanza de *MS-DOS* y lenguaje ensamblador a alumnos de primer curso de las ingenierías técnicas de Informática de Gestión y de Sistemas, los aspectos pedagógicos y un estudio estadístico detallado de los resultados obtenidos.

1.- INTRODUCCIÓN.

Hoy en día los lenguajes de alto nivel, orientados a objetos o lenguajes visuales, separan cada vez más al programador del hardware real de la máquina. Con dichos lenguajes, los desarrollos son mucho más rápidos, y más baratos de diseñar y mantener, facilitando también el prototipado de cara a los clientes de las futuras aplicaciones.

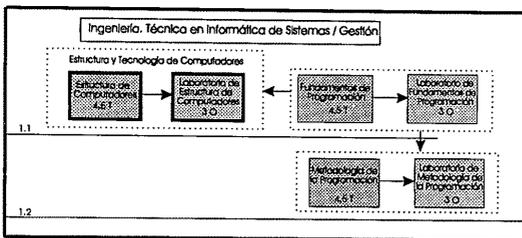
Los lenguajes visuales de programación y orientados a objetos tienen como característica principal la portabilidad de los programas de unas máquinas a otras, ya que ocultan o simplifican las características propias del hardware sobre el que se ejecuta el software final.

Cualquier lenguaje de alto nivel siempre necesitará compiladores, empleará librerías, etc. que requieren del lenguaje ensamblador. Además, existen aplicaciones que no se pueden programar empleando lenguajes de alto nivel, como por ejemplo las diseñadas para ser empleadas en sistemas microcontrolados. En estos sistemas se debe tener una visión muy cercana al hardware sobre el que se ejecutará la aplicación final, por lo que se tiene que acudir a lenguajes que permitan manejar los elementos hardware del sistema de forma directa. Aquí subyace el interés del lenguaje ensamblador.

El lenguaje ensamblador, además, proporciona al usuario / programador una visión clara de las particularidades del hardware sobre el que se ejecutará la aplicación. Esta característica hace que los programas desarrollados en lenguaje ensamblador no sean portables. Es la cercanía al hardware lo que hace que se utilice en la enseñanza de arquitectura.

La asignatura sobre la que se ha realizado el estudio, *Laboratorio de Estructura de Computadores*, es una asignatura obligatoria de primer curso de Informática. Los alumnos llegan a ella con pocos o ningún conocimiento sobre hardware de ordenadores o sobre lenguajes de programación, lo que complica aún más la enseñanza.

El plan de estudios actual no consigue solventar esta situación, pues como se muestra en el siguiente gráfico, la relación de asignaturas de *Programación* y de *Estructura de Computadores*, se imparten a la vez que dicho laboratorio, e incluso dándose el caso, como es el de *Metodología de Programación* de desarrollarse en el siguiente cuatrimestre.



2.- TEMARIO DE LA ASIGNATURA.

La asignatura de *Laboratorio de Estructura de Computadores* (B.O.E. 24-5-95) es una asignatura de primer curso y primer semestre.

El objetivo de la asignatura es dar a los alumnos unos conocimientos básicos sobre la estructura de un computador, mediante la explicación de un sistema operativo sencillo, como es el *MS-DOS*, basado en órdenes, y la programación en lenguaje ensamblador de un microprocesador comercial, el *i80x86*, (que pueden tener incluso en su casa).

Cada una de estos módulos se desarrolla en:

a) Conceptos de sistema operativo.

El sistema operativo empleado es el *MS-DOS*, pues introduce de forma sencilla las funciones de un sistema operativo. No obstante, se hace referencia a los nuevos sistemas operativos como Windows 98, Windows 2.000 y Linux, para enriquecer su formación en sistemas operativos.

Se le da al alumno una visión de cómo almacenar la información en archivos y directorios, cómo manejarse por medio de las órdenes de *MS-DOS* y cómo programar archivos de proceso por lotes que faciliten y mejoren las limitaciones de algunas órdenes.

b) Conceptos de arquitectura mediante el empleo del lenguaje ensamblador.

Es la parte más difícil de impartir, ya que los alumnos carecen en su mayoría de conocimientos de arquitectura de computadores por un lado, y por otro de lenguajes de programación.

Mediante el lenguaje ensamblador se consiguen los siguientes objetivos:

- se muestra el concepto de unidad aritmético-lógica y se enfatiza el sistema de representación de complemento a dos, puesto que el i80x86 opera matemáticamente en dicho formato;
- se les muestra también la estructura del i80x86, los registros que lo forman y las características de cada uno de ellos, registros generales, de segmento, punteros de pila, contador de programa, etc;
- se introduce el concepto de segmentación de memoria y como influye en los diferentes modos de direccionamiento a memoria;
- finalmente se les imparte un repertorio básico de instrucciones, el concepto de interrupciones software y algunas funciones con los que desarrollar las prácticas de la asignatura.

Al finalizar, el alumno habrá comprobado de forma práctica alguna de las características de una arquitectura *von Neumann*: unidad aritmético-lógica, sistemas de memoria, entrada salida, etc.

3.- PEDAGOGÍA EMPLEADA.

Durante los cursos **96-97**, **97-98** y **98-99**, el método empleado fue la presentación mediante transparencias de cada uno de los temas y la explicación de todos aquellos conceptos necesarios para realizar con éxito cada una de las prácticas de laboratorio. A continuación, se realizaba una breve descripción de la práctica, y alguna indicación en cuanto a cómo utilizar los conceptos explicados para la realización de la misma.

Los alumnos, en grupos de dos, realizaban las prácticas, y documentaban el proceso con una memoria, que entregaban al profesor, y que era devuelta con indicaciones sobre errores, posibles mejoras, etc. de forma que le fuera útil para terminar de fijar los conocimientos, y además con vistas al examen.

En el curso **99-00** el método anterior se completó con una serie de actividades previas a cada práctica, que guiaban al alumno en la realización de las mismas y facilitaban la comprensión de la complejidad teórica de dichas prácticas, trasladando parte de los contenidos conceptuales al proceso de realización de actividades. De esta forma el alumno, en lugar de recibir y poner en práctica todos los conceptos en bloque, los iba descubriendo por sí mismo (de una forma tutelada) con lo que se le exigía una actitud más activa. Redundando en un mejor aprendizaje y asimilación.

4.- MÉTODOS DE EVALUACIÓN EMPLEADOS.

A lo largo de los últimos cuatro años, nos hemos planteado cómo evaluar una asignatura de laboratorio, qué peso dar a cada uno de los módulos. Es claro, que si a los alumnos se les exige la entrega de prácticas, ese trabajo debería ser tenido en cuenta en la calificación final de la asignatura.

Durante los cursos **96-97** y **97-98**, el método de evaluación consistía en que los alumnos podían obtener un punto de la calificación final mediante la entrega y superación de las prácticas del laboratorio, pero finalmente, existía un examen, que debido al gran número de alumnos existentes en un curso de primero, debía ser por escrito, aportando así los nueve puntos restantes de la calificación final. No obstante, aunque el método parezca *duro*, se verá en el capítulo 5 que los resultados fueron buenos.

Durante el curso **98-99**, nos planteamos darle un mayor valor a las prácticas de laboratorio, pudiendo obtener hasta tres puntos de la calificación final de la asignatura, aportando los siete puntos restantes un examen por escrito. Como se verá en el capítulo 5 los resultados fueron desastrosos.

Finalmente en el curso **99-00**, se optó por que la evaluación fuese eminentemente práctica. Los alumnos podían obtener el 100 % de la nota final mediante la entrega y superación de las prácticas. Entre todas las prácticas, había algunas que debían ser superadas obligatoriamente, con calificación de *apta* o *no apta*. El resto de las prácticas contribuían a la calificación final valorándose la calidad del desarrollo de las mismas. La forma de superación de cada una de las prácticas era oral, en unos plazos fijados desde el comienzo del curso, y de manera individual por cada uno de los componentes del grupo de prácticas. Para aquellos alumnos que no superasen las prácticas, existía la posibilidad de un examen final en el laboratorio, para superar la asignatura. Como se en el capítulo 5, este método proporcionó muy buenos resultados académicos.

5.- ESTUDIO ESTADÍSTICO DE LOS RESULTADOS ACADÉMICOS.

Las tablas siguientes muestran los estudios estadísticos, ANOVA, test de Kruskal-Wallis y LSD para los resultados de las titulaciones de I.T. Informática de Gestión y de Sistemas respectivamente.

a) I.T. Informática de Gestión.

I.T. Informática de Gestión							
Curso	Pres./ No pres.	Media	Desviación típica	Coeficiente de simetría	Coeficiente de curtosis	Intervalo de confianza al 95%	
						Media	Desviación típica
96-97	56 / 64	5,6	1,71	1,607	-0,811	[5,2 ; 6,1]	[1,44 ; 2,10]
97-98	74 / 104	6,5	2,13	-2,413	0,65269	[6,0 ; 7,0]	[1,83 ; 2,54]
98-99	58 / 134	4,5	2,44	0,349	-1,557	[3,9 ; 5,2]	[2,06 ; 2,99]
99-00	138 / 176	6,2	1,10	-11,692	21,058	[6,0 ; 6,4]	[0,99 ; 1,25]

A continuación veamos los resultados al aplicar diferentes test:

Análisis de la varianza (ANOVA) al 95%					
Fuente	Suma de cuadrados	Df	Media de cuadrados	F-ratio	p-valor
Entre grupos	169,169	3	56,3897	17,88	0
Dentro de grupos	1018,47	323	3,15315		
Total (correlación)	1187,64	326			

Como el p-valor < 0,05 se puede afirmar con el 95% de confianza que existe una diferencia significativa entre las medias de los cuatro años.

Test de Kruskal-Wallis		
Curso	Tamaño muestra	Rango medias
96-97	56	143,7
97-98	74	197,5
98-99	59	105,6
99-00	138	179,2
Estadística del test 38,36 p-v: 0r 2,36E-8		

Como p-valor < 0,05 se puede asegurar que existen diferencias entre las medias.

Test de Fisher de la menor diferencia entre medias (LSD) al 95%			
	Tamaño población	Media	Grupos Homogéneos
Curso 96-97	56	5,6	X
Curso 97-98	74	6,5	X
Curso 98-99	59	4,4	X
Curso 99-00	138	6,2	X
Contraste	Diferencia	+/- límites	
96-97 rente a 97-98	-0,9	0,62	
96-97 rente a 98-99	1,2	0,65	
96-97 rente a 99-00	-0,5	0,55	
97-98 rente a 98-99	2,1	0,61	
97-98 rente a 99-00	0,3	0,50	
98-99 rente a 99-00	-1,7	0,54	

A la vista de los resultados anteriores se puede ver que realmente sí existen diferencias entre las medias de los diferentes cursos.

b) I.T. Informática de Sistemas.

I.T. Informática de Sistemas							
Curso	Tamaño población	Media	Desviación típica	Coeficiente de simetría	Coeficiente de curtosis	Intervalo de confianza al 95%	
						Media	Desviación típica
96-97	56 / 59	5,2	1,92	0,498	-0,377	[4,7 ; 5,8]	[1,61 ; 2,35]
97-98	79 / 97	6,7	1,71	0,311	-1,063	[6,3 ; 7,1]	[1,48 ; 2,03]
98-99	87 / 142	5,2	2,37	-0,97	-1,928	[4,6 ; 5,7]	[2,06 ; 2,79]
99-00	143 / 171	6,4	0,98	-6,857	16,005	[6,2 ; 6,5]	[0,88 ; 1,11]

Análisis de la varianza (ANOVA) al 95%					
Fuente	Suma de cuadrados	Df	Media de cuadrados	F-ratio	p-valor
Entre grupos	150,129	3	50,0431	17,15	0
Dentro de grupos	1053,23	361	2,91753		
Total (correlación)	1203,36	364			

Como el p-valor < 0,05 se puede afirmar con el 95% de confianza que existe una diferencia significativa entre las medias de los cuatro años.

Test de Kruskal-Wallis		
Curso	Tamaño muestra	Rango medias
96-97	56	134,045
97-98	79	216,089
98-99	87	151,839
99-00	143	202,85
Estadística del test 32,7391 p-valor 3,65E-7		

Como p-valor < 0,05 se puede asegurar que existen diferencias entre las medias.

Test de Fisher de la menor diferencia entre media (LSD) al 95%			
Curso	Tam.Poblac	Media	Grupos Homogéneos
Curso 96-97	56	5,2	X
Curso 97-98	79	6,7	X
Curso 98-99	87	5,2	X
Curso 99-00	143	6,4	X
Contraste	Diferencia	+/- límites	
96-97 frente a 97-98	-1,46	0,59	
96-97 frente a 98-99	0,053	0,57	
96-97 frente a 99-00	-1,14	0,53	
97-98 frente a 98-99	1,51	0,52	
97-98 frente a 99-00	0,32	0,47	
98-99 frente a 99-00	-1,20	0,45	

A la vista de los resultados anteriores se puede ver que realmente sí que existe diferencia de medias entre diferentes cursos.

6.- CONCLUSIONES.

La asignatura de *Laboratorio de Estructura de Computadores* es de gran complejidad, ya que el alumno debe enfrentarse a ella sin conocimientos de hardware ni de programación y que deben ser impartidos a la par que la asignatura. De esta forma los profesores se encuentran con la difícil situación de explicar unos conceptos que deberían ser previos a la impartición de la asignatura, junto con los propios de la misma. La razón se encuentra en el plan de estudios, puesto que dichos conceptos se imparten al final de asignaturas que comparten temporalidad con la de *Laboratorio de Estructura de Computadores*.

El estudio estadístico anterior pone en evidencia que el mejor de los métodos empleados para impartir la asignatura de *Laboratorio de Estructura de Computadores* ha sido el seguido durante el curso 99-00, puesto que aunque no existe una diferencia significativa en las medias con el curso 97-98, en comparación, en el año 99-00 el número de alumnos que superan la asignatura, se ha incrementado considerablemente.

Finalmente, debido a las actividades desarrolladas previa a las prácticas, a los profesores que hemos impartido la asignatura nos ha quedado patente el aumento del interés de los alumnos, y como consecuencia una mejora de las calificaciones.

7.- BIBLIOGRAFÍA.

F. Segovia Olmo y J. Beltrán Llera. El aula inteligente. Nuevo horizonte educativo. Editorial Espasa. 1998

F. Remiro Domínguez y A. Martín García. Microprocesadores: el 8088 / 86 Editorial Akal-Biblioteca tecnológica. 1993.

J. Beltrán de Heredia . Lenguaje ensamblador de los 80x86. Editorial Anaya-Multiemdia. 1996.

K. Jamsa. DOS manual de bolsillo. Ed Osborne-McGraw-Hill. 1.989