

TÉCNICAS DE PROGRAMACIÓN EN BAJO NIVEL.
TERCER CURSO DE INGENIERIA EN INFORMÁTICA
PLAN DE ORGANIZACIÓN DOCENTE DE LA ASIGNATURA
CURSO: 1999/2000

PROFESORES: Claudio A. Amaya Rodríguez, Gabriel Jiménez Moreno.

a) TEMARIO:

- Introducción a la programación de sistema (programación en bajo nivel): Perspectiva histórica, características de la programación de sistema. Lenguajes de programación para el software de sistema.
- Herramientas de programación de sistemas: Ensambladores, montadores, depuradores.
- Técnicas de manejo de entrada salida: Puertos de entrada salida, dispositivos mapeados en memoria, sincronización con eventos externos (polling y manejo de interrupciones), DMA, coprocesadores (gráficos, controladores de red, etc...).
- Relaciones entre lenguajes de alto nivel y lenguaje máquina: paso de parámetros, variables compartidas, implementación de tipos de datos, interfase ensamblador – lenguajes de alto nivel (C, Pascal).
- Interfase con el sistema operativo. Escritura de drivers.

b) BIBLIOGRAFIA:

Básica:

- Michael Thorne. “Computer Organization and Assembly Language Programming : For IBM PCs and Compatibles”. Benjamin/Cummings Publishing, 2nd edition, 1993.
- Peter Abel. “Lenguaje Ensamblador y Programación para PC IBM y compatibles”. 3ª edición. Prentice Hall Hispanoamericana. 1996.

Complementaria:

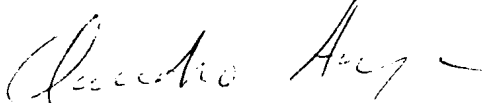
- D.A. Patterson, J.L. Hennessy, “Organización y diseño de computadores. La interfaz hardware/software”. McGraw-Hill, 1995. (versión en castellano de la 1ª edición inglesa, 1993).
- Burke. “Writing Device Drivers: Tutorial & Reference”. Digital Press.
- R. S. Lai. “Writing MS-DOS Device Drivers”. Addison-Wesley, 1994. (2ª edición).
- G. Pajari. “Writing UNIX Device Drivers”. Addison-Wesley, 1997.

c) OBJETIVOS DE LA ASIGNATURA.

Esta asignatura pretende dar una visión de las característica y funciones del software de sistema de un computador típico, así como entender los principios de la programación en bajo nivel y de los soportes para la ejecución y depuración de programas. Para ello se revisará el funcionamiento de los dispositivos de entrada/salida desde el punto de vista del programador. Se usará tanto lenguaje ensamblador como lenguajes de alto nivel para implementar software de manejo de dispositivos. Por último, se tratarán las técnicas que permiten el desarrollo de drivers de dispositivos, es decir, integrar el software de manejo de dispositivos de entrada/salida dentro del nucleo de sistema operativo.

d) PRACTICAS.

Se proponen tres prácticas. Sin embargo, las prácticas y su contenido pueden modificarse dependiendo de la disponibilidad de profesorado y material de laboratorio. En principio se plantean las siguientes:



- Desarrollo de un programa de manejo del puerto de impresora.
- Desarrollo de programas para el manejo de placa digitalizadora de video basada en el chip BT848.
- Desarrollo de driver para BT848 en el sistema operativo DOS o UNIX o RTX.

e) EVALUACION.

Por cada convocatoria habrá un examen que constará de dos partes, con un breve descanso entre ambas. En la primera parte, predominantemente teórica, no se podrá consultar ninguna documentación. Su duración será de aproximadamente 45-60 minutos, y su peso sobre la puntuación del examen será del orden del 30%, aunque su valor concreto se indicará en el mismo enunciado. La segunda parte tendrá un contenido práctico, durará unas 2,5-3 horas, y se podrá consultar aquella documentación que el alumno crea conveniente. También se indicará el mínimo a obtener en cada parte para superar la prueba.

f) TUTORIAS.

Esta actividad se realiza a petición del alumno. Los profesores harán público su horario de tutorías, durante el cual la participación del alumno no precisa de una petición previa. El profesor prestará atención al alumno en todas las cuestiones que conciernen a la asignatura, resolviendo las posibles dudas. Esta actividad no debe considerarse una "clase particular", por lo que se limitará a tratar aspectos previamente trabajados por el alumno.

Ed. Gabriel Jiménez