

## PROGRAMA DE LA ASIGNATURA

Curso académico: 2013/2014

Identificación y características de la asignatura				
Código	501279		Créditos ECTS	6
Denominación (español)	Administración y Organización de Computadores			
Denominación (inglés)	Management and Organization of Computers			
Titulaciones	Grado en Ingeniería Informática en Ingeniería de Computadores Grado en Ingeniería Informática en Ingeniería del Software			
Centro	Escuela Politécnica			
Semestre	3º	Carácter	Obligatorio	
Módulo	Común a la Rama de Informática			
Materia	Ingeniería de Computadores			
Profesor/es				
Nombre	Despacho	Correo-e	Página web	
Pilar Bachiller Burgos	18	<a href="mailto:pilarb@unex.es">pilarb@unex.es</a>	<a href="http://robolab.unex.es">http://robolab.unex.es</a>	
José Miguel Martínez Candela	1	<a href="mailto:josemmar@unex.es">josemmar@unex.es</a>		
Francisco M. Andrés Hernández	17	<a href="mailto:pacoan@unex.es">pacoan@unex.es</a>		
Área de conocimiento	Arquitectura y Tecnología de Computadores			
Departamento	Tecnología de los Computadores y de las Comunicaciones			
Profesor coordinador (si hay más de uno)	Pilar Bachiller Burgos			
Competencias				
<p><b>CB1:</b> Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.</p>				
<p><b>CB2:</b> Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.</p>				
<p><b>CB3:</b> Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.</p>				
<p><b>CB4:</b> Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.</p>				
<p><b>CB5:</b> Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.</p>				
<p><b>CI04:</b> Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.</p>				
<p><b>CI05:</b> Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.</p>				

**CI06:** Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos.

**CT02:** Habilidades de gestión de recursos de información.

**CT03:** Capacidad para resolver problemas.

### OBJETIVOS DE APRENDIZAJE

- ✓ Conocer la organización de un computador desde el punto de vista de su programación de bajo nivel.
- ✓ Conocer las generalidades del nivel ISA de un computador, estudiando casos de máquinas RISC y CISC.
- ✓ Conocer el modelo de programación de la familia de microprocesadores de Intel.
- ✓ Desarrollar programas en lenguaje ensamblador para la arquitectura IA32.
- ✓ Identificar las transformaciones entre construcciones de código escritas en un lenguaje de alto nivel y su equivalente a nivel de código máquina.
- ✓ Conocer los componentes de procesamiento matemático de los procesadores de Intel y saber utilizar la tecnología disponible en el desarrollo de programas de bajo nivel.
- ✓ Saber integrar código escrito en lenguaje ensamblador con código escrito en lenguaje de alto nivel.
- ✓ Conocer y saber utilizar herramientas de administración del sistema operativo.

### RESULTADOS DE APRENDIZAJE

- ✓ Comprende la organización de un computador desde el punto de vista del programador en lenguaje máquina y ensamblador, conociendo las distintas alternativas para el conjunto de instrucciones, los formatos de instrucción y modos de direccionamiento.
- ✓ Utiliza eficientemente el sistema operativo y conoce las herramientas que lo acompañan y que permiten dar respuesta a diferentes problemas.
- ✓ Conoce y aplica en actividades de nivel medio las competencias transversales fundamentales de la profesión.

### Temas y contenidos

#### Breve descripción del contenido

Organización de un computador desde el punto de vista del programador del lenguaje máquina y ensamblador. Arquitecturas del Conjunto de Instrucciones (ISA), formatos de instrucción y modos de direccionamiento. Administración de sistemas operativos.

#### Temario de la asignatura

Denominación del tema 1: Conceptos generales de computadores

Contenidos del tema 1:

- 1.1. Arquitecturas básicas
- 1.2. Coste y rendimiento

<ul style="list-style-type: none"> <li>1.3. Introducción al paralelismo</li> <li>1.4. Maquina cableada y microprogramada</li> <li>1.5. Interrupciones</li> <li>1.6. Memoria</li> </ul>
<p>Denominación del tema 2: Introducción al sistema operativo Linux</p> <p>Contenidos del tema 2:</p> <ul style="list-style-type: none"> <li>2.1. Conceptos básicos</li> <li>2.2. Instalación</li> <li>2.3. Entornos gráficos</li> <li>2.4. El sistema de ficheros</li> <li>2.5. Comandos</li> </ul>
<p>Denominación del tema 3: Arquitectura del conjunto de instrucciones (ISA)</p> <p>Contenidos del tema 3:</p> <ul style="list-style-type: none"> <li>3.1. Generalidades</li> <li>3.2. Tipos de datos</li> <li>3.3. Modos de direccionamiento</li> <li>3.4. Formatos de instrucciones</li> <li>3.5. Microprocesadores RISC y CISC</li> <li>3.6. Nivel ISA de microprocesadores RISC</li> <li>3.7. Nivel ISA de microprocesadores CISC</li> </ul>
<p>Denominación del tema 4: Familia de microprocesadores de Intel</p> <p>Contenidos del tema 4:</p> <ul style="list-style-type: none"> <li>4.1. Generaciones</li> <li>4.2. Características de la arquitectura</li> <li>4.3. Modos de direccionamiento</li> <li>4.4. Repertorio de instrucciones</li> <li>4.5. Intel versus AMD</li> </ul>
<p>Denominación del tema 5: Representación de programas a nivel de máquina</p> <p>Contenidos del tema 5:</p> <ul style="list-style-type: none"> <li>5.1 Representación de datos</li> <li>5.2 Estructuras de control</li> <li>5.3 Llamadas a procedimientos. Paso de parámetros</li> <li>5.4 Gestión del bloque de activación</li> <li>5.5 Ensamblador en línea</li> </ul>
<p>Denominación del tema 6: Aspectos avanzados de los microprocesadores de Intel</p> <p>Contenidos del tema 6:</p> <ul style="list-style-type: none"> <li>6.1. Coprocesador matemático</li> <li>6.2. MMX</li> <li>6.3. SSE</li> </ul>
<p style="text-align: center;"><b>Organización de las prácticas</b></p> <p><u>Introducción al sistema operativo Linux (sesiones 1-3)</u></p> <ul style="list-style-type: none"> <li>▣ <u>Sesión 1</u> (1 h): <ul style="list-style-type: none"> <li>○ Conceptos básicos</li> </ul> </li> </ul>

- Instalación, arranque y parada

▣ Sesión 2 (1,5 h):

- Entornos de escritorio
- El sistema de ficheros
- Comandos

▣ Sesión 3 (1,5 h):

- Comandos
- Ejercicios de comandos en Linux.

Programación en ensamblador de la arquitectura IA32 (sesiones 4-9)

▣ Sesión 4 (1,5 h):

- Características generales de la arquitectura IA32
- Sintaxis de un programa en lenguaje ensamblador

▣ Sesión 5 (1,5 h):

- Generación de código ejecutable
- Uso de librerías
- Depuración de código (lo que dé tiempo).

En las sesiones de la 6 a la 9 (1,5 h por sesión) se analizará un código de ejemplo relacionado con los aspectos concretos a tratar durante la sesión. A continuación, se propondrá un ejercicio relacionado que deberán desarrollar y entregar al final de la sesión.

Ensamblador en línea con gcc (sesiones 10 y 11)

Las sesiones 10 y 11 (1,5 h por sesión) se dedicarán al desarrollo de la práctica final de la asignatura, consistente en un proyecto de programación en ensamblador/C++.

**Actividades formativas**

Horas de trabajo del alumno por tema		Presencial		Actividad de seguimiento	No presencial
Tema	Total	GG	SL	TP	EP
1	18,5	8	0	0	10,5
2	11	0	4	0	7
3	23	9	0	0	14
4	39	8	4	1	26
5	34,5	6,5	5	1	22
6	24	6	3	0	15
<b>Evaluación del conjunto</b>	<b>150</b>	<b>37,5</b>	<b>16</b>	<b>2</b>	<b>94,5</b>

GG: Grupo Grande (100 estudiantes).

SL: Seminario/Laboratorio (prácticas clínicas hospitalarias = 7 estudiantes; prácticas laboratorio o campo = 15; prácticas sala ordenador o laboratorio de idiomas = 30, clases problemas o seminarios o casos prácticos = 40).

TP: Tutorías Programadas (seguimiento docente, tipo tutorías ECTS).

EP: Estudio personal, trabajos individuales o en grupo, y lectura de bibliografía.

**Presenciales en grupo grande**

Se realizarán clases expositivas para el desarrollo de los contenidos teóricos. Asimismo, se realizarán clases de explicación y resolución de ejercicios y problemas. Por último, se llevarán a cabo actividades individuales o en grupo destinadas a aplicar los conceptos expuestos a la resolución de problemas.

**Presenciales en laboratorio**

Se dedicarán varias sesiones prácticas a conocer el sistema operativo y las herramientas software que se utilizarán a lo largo de la asignatura. Se propondrán problemas de programación que habrá que resolver durante la sesión. Se planteará una práctica de programación que se desarrollará tanto dentro como fuera del laboratorio, realizando actividades de seguimiento durante las sesiones prácticas correspondientes.

**Tutorías programadas**

Se utilizarán para el seguimiento de las actividades planteadas a lo largo del semestre y la evaluación individual de los objetivos alcanzados.

**No presencial**

Las actividades no presenciales necesarias para alcanzar los objetivos de aprendizaje se resumen en las siguientes: estudio individual, búsqueda de información, desarrollo de programas, resolución de problemas, realización de cuestionarios.

**Sistemas de evaluación**

Instrumentos de evaluación

Se utilizarán los siguientes instrumentos de evaluación:

- ✓ Portafolio de actividades: conjunto de actividades realizadas por el estudiante a lo largo del semestre (no recuperable). Se realizarán entre 2 y 5 actividades de este tipo a lo largo del curso.
- ✓ Prácticas de programación: resolución de problemas de programación de bajo nivel de la arquitectura IA32 aplicando los distintos conocimientos adquiridos a lo largo de la asignatura.
- ✓ Prueba escrita: preguntas de tipo test y resolución de problemas.

Criterios de evaluación

Para superar la asignatura será necesario obtener una calificación superior o igual a 5 (sobre 10) en las prácticas de programación y en la prueba escrita. La evaluación de las prácticas de programación considerará la resolución de ejercicios propuestos durante las sesiones (30%), así como la realización de una práctica final (70%). Para aprobar la práctica final será necesario superar un examen de modificación propuesto.

La prueba escrita consistirá en una prueba de tipo test y un examen de problemas. Para superar la prueba escrita será necesario obtener al menos un 4 en cada parte. Tras superar esta nota de corte, la nota final del bloque se obtendrá como el 25% de la nota de la prueba de tipo test más el 75% de la nota del examen de problemas.

Cada nota asociada con los bloques de prácticas y de prueba escrita supondrá el 40% de la nota final. El 20% restante estará asociado con el portafolio de actividades. Así, una vez superados los bloques de prácticas de programación y prueba escrita, la nota final del estudiante se obtendrá de la siguiente forma:

$$\text{Nota\_Final} = 0,2 * \text{Nota\_Portafolio} + 0,4 * \text{Nota\_Prácticas} + 0,4 * \text{Nota\_Prueba\_Escrita}$$

En otro caso, la calificación se calculará según la siguiente tabla:

<b>Prácticas Programación</b>	NP	-	<5	≥5	<5
<b>Prueba Escrita</b>	-	NP	<5	<5	≥5
<b>Nota Final</b>	NP	NP	1	2	2

NP: No Presentado

La nota en cada bloque superado se guardará durante todas las convocatorias del curso.

### Bibliografía y otros recursos

- [Angulo03] J.M. Angulo, J.L. Gutiérrez e I. Angulo. *Arquitectura de microprocesadores. Los Pentium a fondo*. Paraninfo, 2003.
- [Brey09] B.B. Brey. *Intel Microprocessors. Architecture, Programming and Interfacing*. Pearson – Prentice Hall, 2009.
- [Bryant10] R.E. Bryant, D.R. O'Hallaron. *Computer Systems: a Programmer's Perspective*. Pearson Education, 2010.
- [Charte09] F. Charte. *Ensamblador (edición 2009)*. Anaya multimedia, 2009.
- [Garcia00] J. García de Jalón, I. Aguinaga, A. Mora. *Aprenda LINUX como si estuviera en primero*. Universidad de Navarra, 2000.
- [Hennessy93] J.L. Hennessy y D.A. Patterson. *Arquitectura de computadores. Un enfoque cuantitativo*. McGraw-Hill, 1993
- [Pardo08] A. Pardo. *Programación en ensamblador de la arquitectura IA-32*. Universidad Carlos III de Madrid, 2008.
- [Tanenbaum00] A.S. Tanenbaum. *Organización de computadores. Un enfoque estructurado*. Pearson Educación, 2000.
- [Ubuntu] *Manual de Ubuntu 9.04*.
- [Ujaldón03] M. Ujaldón. *Arquitectura del PC. Volumen I: Microprocesadores*. Editorial Ciencia-3, 2003.

### Horario de tutorías

Tutorías Programadas: *se fijarán al comienzo del curso en coordinación con las restantes asignaturas del semestre.*

Tutorías de libre acceso: *se publicarán en la web del Centro y en la puerta del despacho del profesor en los plazos previstos por la Normativa vigente de Tutorías.*

### Recomendaciones

- ✓ Se recomienda la asistencia a las clases teóricas y prácticas, así como la realización de las actividades planteadas a lo largo de semestre.
- ✓ Se recomienda el acceso regular al aula virtual de la asignatura.
- ✓ Se recomienda una dedicación continuada a la asignatura que permita completar las horas en el aula con la comprensión de los conceptos tratados y la resolución autónoma de problemas.