

## PROGRAMA DE LA ASIGNATURA

**Curso académico: 2017/2018**

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="https://robolab.unex.es/">https://robolab.unex.es/</a> <a href="http://epcc.unex.es/">http://epcc.unex.es/</a>	
José Miguel Martínez Candela	1	<a href="mailto:josemmar@unex.es">josemmar@unex.es</a>	<a href="http://epcc.unex.es/">http://epcc.unex.es/</a>	
Francisco M. Andrés Hernández	17	<a href="mailto:pacoan@unex.es">pacoan@unex.es</a>	<a href="http://epcc.unex.es/">http://epcc.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>CG04:</b> Capacidad para definir, evaluar y seleccionar plataformas hardware y software para el desarrollo y la ejecución de sistemas, servicios y aplicaciones informáticas, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 del anexo II de la resolución de la Secretaría General de Universidades de 8 de junio de 2009 (BOE de 4 de agosto de 2009) para la tecnología específica de Ingeniería del Software e Ingeniería de Computadores.</p>				
<p><b>CG06:</b> Capacidad para concebir y desarrollar sistemas o arquitecturas informáticas centralizadas o distribuidas integrando hardware, software y redes, de acuerdo con los conocimientos adquiridos según lo establecido en el apartado 5 del anexo II de la resolución antes mencionada.</p>				
<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>				

**CB3:** 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.

**CB4:** Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

**CB5:** Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

**CI04:** Capacidad para elaborar el pliego de condiciones técnicas de una instalación informática que cumpla los estándares y normativas vigentes.

**CI05:** Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.

**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.

### **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
- 1.3. Introducción al paralelismo
- 1.4. Máquina cableada y microprogramada
- 1.5. Interrupciones
- 1.6. Memoria

Denominación del tema 2: Introducción al sistema operativo Linux

Contenidos del tema 2:

- 2.1. Conceptos básicos
- 2.2. Instalación
- 2.3. Entornos gráficos
- 2.4. El sistema de ficheros
- 2.5. Comandos

Denominación del tema 3: Arquitectura del conjunto de instrucciones (ISA)

Contenidos del tema 3:

- 3.1. Generalidades
- 3.2. Tipos de datos
- 3.3. Modos de direccionamiento
- 3.4. Formatos de instrucciones
- 3.5. Microprocesadores RISC y CISC
- 3.6. Nivel ISA de microprocesadores RISC
- 3.7. Nivel ISA de microprocesadores CISC

Denominación del tema 4: Familia de microprocesadores de Intel

Contenidos del tema 4:

- 4.1. Generaciones
- 4.2. Características de la arquitectura
- 4.3. Modos de direccionamiento
- 4.4. Repertorio de instrucciones
- 4.5. Intel versus AMD

Denominación del tema 5: Representación de programas a nivel de máquina

Contenidos del tema 5:

- 5.1. Representación de datos
- 5.2. Estructuras de control
- 5.3. Llamadas a procedimientos. Paso de parámetros
- 5.4. Gestión del bloque de activación
- 5.5. Ensamblador en línea

Denominación del tema 6: Aspectos avanzados de los microprocesadores de Intel

Contenidos del tema 6:

- 6.1. Coprocesador matemático
- 6.2. MMX

## 6.3. SSE

### Organización de las prácticas

#### Introducción al sistema operativo Linux (sesiones 1-3)

- Sesión 1 (1,5 h):
  - Conceptos básicos
  - Instalación, arranque y parada
  - Entornos de escritorio
  - El sistema de ficheros
- Sesión 2 (1,5 h):
  - Comandos
- Sesión 3 (1 h):
  - 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>	150	37,5	16	2	94,5

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.

### Metodologías docentes

- ✓ En Clases teórico-prácticas en el aula, 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.
- ✓ En sesiones de 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.
- ✓ En tutorías programadas. individuales o en grupos pequeños se realizará el seguimiento de las actividades planteadas a lo largo del semestre y la evaluación individual de los objetivos alcanzados.
- ✓ Realización de actividades, trabajos y estudio por parte del estudiante, de manera autónoma, 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.

### Resultados de aprendizaje

- ✓ Conoce la estructura de los computadores desde el punto de vista de las distintas técnicas que se pueden utilizar para implementar la unidad central de proceso (CPU), con especial énfasis en la técnica de segmentación. Además también sabe aplicar correctamente diversas medidas de rendimiento.
- ✓ 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.
- ✓ Conoce y aplica en actividades de nivel medio las competencias

transversales fundamentales de la profesión.

## **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. Se realizarán entre 2 y 5 actividades de este tipo a lo largo del curso.
- ✓ Prácticas de programación: resolución de problemas de administración de sistemas operativos y de programación de bajo nivel de la arquitectura IA32 aplicando los distintos conocimientos adquiridos a lo largo de la asignatura.
- ✓ Pruebas escritas: preguntas de tipo test, preguntas cortas, resolución de problemas, etc.

### **Criterios de evaluación**

Tal y como se contempla en la 'Normativa de evaluación de los resultados de aprendizaje y de las competencias adquiridas por el alumnado en las titulaciones oficiales de Universidad de Extremadura' vigente (DOE 12 de diciembre de 2016), esta asignatura puede superarse siguiente el sistema de evaluación continua o mediante una prueba final de carácter global.

Como se indica en esta normativa, "la elección entre el sistema de evaluación continua o el sistema de evaluación con una única prueba final de carácter global corresponde al estudiante durante las tres primeras semanas de cada semestre."

El estudiante comunicará por escrito a los profesores de la asignatura el tipo de evaluación elegido utilizando el modelo que se encontrará en el aula virtual. Si un estudiante no comunica el tipo de evaluación elegido en el plazo indicado, se supondrá que opta por la evaluación continua.

### **Evaluación continua**

Para superar la asignatura por el sistema de evaluación continua será necesario obtener una calificación superior o igual a 5 (sobre 10) en las prácticas de programación y en la prueba escrita de evaluación del temario teórico que se realizará en las convocatorias oficiales. La primera parte de dicha prueba podrá convalidarse con el resultado de un examen parcial que tendrá lugar durante la primera mitad del semestre, siempre que dicho resultado supere el aprobado. Esta opción sólo podrá aplicarse en la convocatoria de enero.

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.

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 (no recuperable). 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.

### **Evaluación por prueba final global**

Para superar la asignatura será necesario obtener una nota superior o igual a 5 en los bloques de teoría y prácticas de manera individual.

Para superar el bloque de prácticas será necesario realizar la práctica final y superar un examen de modificación propuesto. La nota obtenida de dicha práctica supondrá el 70% de la nota de todo el bloque de prácticas. El 30% restante se obtendrá de un examen de prácticas en el que se evaluarán contenidos no incluidos en la práctica final. Si la parte asociada con la práctica final no se supera, la calificación final de todo el bloque será de 2.

La evaluación del bloque de teoría se realizará mediante una prueba de tipo test y un examen de problemas. Este último será común para todos los estudiantes con independencia del tipo de evaluación al que se hayan acogido. Para superar el bloque de teoría, será necesario obtener una calificación mayor o igual a 5 en el examen de problemas. Tras superar esta nota, la nota final del bloque se obtendrá como el 30% de la nota de la prueba tipo test más el 70% de la nota del examen de problemas. Si la nota del examen de problemas fuera inferior a 5, la nota de todo el bloque de teoría será de 2.

Una vez superados los bloques de teoría y prácticas, la nota final del estudiante se obtendrá de la siguiente forma:

$$\text{Nota\_Final} = 0,5*\text{Nota\_Teoría} + 0,5*\text{Nota\_Prácticas}$$

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

<b>Nota Teoría</b>	NP	-	<5	≥5	<5
--------------------	----	---	----	----	----

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

**Bibliografía (básica y complementaria)**

[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.

[Patterson14] D.A. Patterson y J.L. Hennessy. *Computer Organization and Design (5th edition). The Hardware/Software Interface*. Morgan Kaufmann (Elsevier), 2014

[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.

**Otros recursos y materiales docentes complementarios**

El aula virtual de la asignatura contiene recursos relacionados con todos los temas del temario teórico, así como el material necesario para el desarrollo de la parte práctica.

**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.