



Computación Para Ingeniería

Guía de Aprendizaje – Información al estudiante

1. Datos Descriptivos

Asignatura	Computación para Ingeniería
Materia	Computación para Ingeniería
Departamento responsable	Lenguajes y Sistemas Informáticos en Ingeniería del Software
Créditos ECTS	6
Carácter	Obligatorio
Titulación	Máster Universitario en Ingeniería Informática por la Universidad Politécnica de Madrid
Curso	1º
Especialidad	No aplica

Curso académico	2013-2014
Semestre en que se imparte	Segundo (febrero a junio)
Semestre principal	Feb.-Junio
Idioma en que se imparte	Español
Página Web	



2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
Vicente Martín Ayuso (Coord.)	5209	vicente@fi.upm.es
Jose María Peña	4201	jmpena@fi.upm.es
María S. Pérez	4204	mperez@fi.upm.es
Antonio García Dopico	4104	dopico@fi.upm.es
Fernando Pérez	4201	fperez@fi.upm.es
Juan Robles Santamarta	5201	jrobles@fi.upm.es

3. Conocimientos previos requeridos para poder seguir con normalidad la asignatura

Asignaturas superadas	<ul style="list-style-type: none">•
Otros resultados de aprendizaje necesarios	<ul style="list-style-type: none">•



4. Objetivos de Aprendizaje

COMPETENCIAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN		
Código	Competencia	Nivel
CG1	Que los estudiantes sepan aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios (o multidisciplinares) relacionados con su área de estudio.	A
CG9	Aplicación de los métodos de resolución de problemas más recientes o innovadores y que puedan implicar el uso de otras disciplinas.	P
CG13	Apreciación de los límites del conocimiento actual y de la aplicación práctica de la tecnología más reciente.	P
CG15	Capacidad para contribuir al desarrollo futuro de la informática.	P
CE-4	Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.	P
CE-9	Capacidad para diseñar y evaluar sistemas operativos y servidores, y aplicaciones y sistemas basados en computación distribuida.	P
CE-10	Capacidad para comprender y poder aplicar conocimientos avanzados de computación de altas prestaciones y métodos numéricos o computacionales a problemas de ingeniería.	A

LEYENDA: Nivel de competencia: conocimiento (1), comprensión (2), aplicación (3) y análisis y síntesis (4),



RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1	Aplicar técnicas y herramientas de computación de alto rendimiento para la solución de problemas prácticos	CG1, CG9, CE10	A
RA2	Aplicar algoritmos numéricos al modelado de problemas prácticos	CG1, CG9, CE10	A
RA3	Relacionar las necesidades de los algoritmos numéricos en el modelado de problemas con su implementación práctica en hardware/software de alto rendimiento	CG13, CG15, CE4, CE9, CE10	A

5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relacionado con RA
I1	Implementar algunos de los algoritmos numéricos más utilizados en la práctica de la ingeniería informática.	RA2
I2	Analizar la relación de los algoritmos numéricos con la infraestructura HW y SW para su correcta implementación en sistemas de alto rendimiento.	RA1, RA2, RA3
I3	Estudiar el coste HW (memoria, CPU, comunicaciones, almacenamiento, etc.) de algoritmos numéricos en casos prácticos	RA2
I4	Estudiar la implementación de algoritmos numéricos en términos de su rendimiento y coste en una implementación práctica en sistemas de alto rendimiento.	RA1,RA2, RA3
I5	Aplicar las herramientas HW y SW de alto rendimiento necesarias para el uso práctico de los algoritmos numéricos.	RA1,RA2, RA3
I5		
I6		

EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Primer Proyecto: Definición y defensa del problema a resolver	Semana 3	Aula habitual (con ordenador conectado a la red)	15.00%



EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Primer Proyecto: Defensa de la solución implementada	Semana 5	Aula habitual (con ordenador conectado a la red)	25.00%
Primer Proyecto: Memoria escrita	Semana 5		10.00%
Segundo Proyecto: Definición y defensa del problema a resolver	Semana 6	Aula habitual (con ordenador conectado a la red)	15.00%
Segundo Proyecto: Defensa de la solución implementada	Semana 8	Aula habitual (con ordenador conectado a la red)	25.00%
Segundo Proyecto: Memoria escrita	Semana Exámenes		10.00%
			Total: 100%



CRITERIOS DE CALIFICACIÓN

El método de evaluación normal de la asignatura es el de evaluación continua. Conforme a la normativa UPM, se admite también el método de evaluación única para aquellos alumnos que así lo deseen. Para ello, deberán solicitarlo por escrito al coordinador de la asignatura en un plazo no superior a 30 días tras el inicio de las clases.

Evaluación continua:

Para poder ser evaluado de manera continua se requiere una asistencia mínima al 70% a las actividades de la asignatura.

La calificación se obtendrá mediante pruebas orales, trabajos/proyectos e informes/memorias de los trabajos prácticos.

Las pruebas se organizan entorno a 2 proyectos que el alumno deberá realizar durante el transcurso de la asignatura. Una terminará en la primera mitad y otra en la segunda.

El proyecto de la primera mitad versará sobre algoritmos numéricos y la solución de problemas utilizando librerías que implementen dichos métodos.

El de la segunda mitad versará sobre la implementación de algoritmos numéricos utilizando alguno de los paradigmas de programación estudiados.

Cada proyecto será evaluado en tres fases. La primera consistirá en la definición y defensa del problema propuesto. Esta defensa se realizará mediante presentación oral. La segunda fase, realizada a la terminación del proyecto consistirá en la exposición y defensa del trabajo. La tercera será la evaluación de la memoria escrita presentada.

El valor de cada uno de los dos proyectos es el mismo. El valor de cada parte es de 30%-50%-20% respectivamente para un total de 100% por proyecto.

Evaluación única: Acorde a la normativa de exámenes (artículo 20.2) de la universidad, se permite una evaluación única, no continua, para aquellas alumnos que así lo soliciten. Los alumnos que lo deseen deberán solicitarlo por escrito al coordinador de la asignatura en un plazo no superior a 30 días tras el inicio de las clases.

Esta evaluación consistirá en un examen de implementación de algoritmos numéricos y la solución de problemas propuestos. Se realizará en las fechas



POLITÉCNICA

UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
Campus de Vicarganda
Calle de Velázquez, 28560 Madrid

CRITERIOS DE CALIFICACIÓN

establecidas por jefatura de estudios.

Examen Extraordinario de Julio: Acorde a la normativa de la universidad, se establece una convocatoria extraordinaria que consiste en un examen de implementación de algoritmos numéricos y la solución de problemas propuestos. Se realizará en las fechas establecidas por jefatura de estudios.



6. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicadores Relacionados
Tema 1: Algorítmica Numérica	Algoritmos y prestaciones del álgebra lineal numérica: representación y errores, resolución y condicionamiento de grandes sistemas lineales, algoritmos de búsqueda en la red (PageRank de Google),...	I1,I2,I3
Tema 2: Programación y Librerías Numéricas	Optimización secuencial, Programación paralela (OpenMP y MPI). Librerías y paquetes matemáticos (BLAS, PBLAS, LAPACK, ScaLAPACK, FFTPACK, NASTRAN...)	I1,I2,I3,I4
Tema 3: Arquitecturas y Sistemas	Arquitecturas paralelas, tipologías, redes de interconexión. Servicios: E/S paralela.	I2,I3,I4

7. Breve descripción de las modalidades organizativas utilizadas y de los métodos de enseñanza empleados

Tabla 7. Modalidades organizativas de la enseñanza







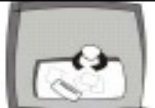
MODALIDADES ORGANIZATIVAS DE LA ENSEÑANZA		
Escenario	Modalidad	Finalidad
	Clases Teóricas	<i>Hablar a los estudiantes</i>
	Seminarios-Talleres	<i>Construir conocimiento a través de la interacción y la actividad de los estudiantes</i>
	Clases Prácticas	<i>Mostrar a los estudiantes cómo deben actuar</i>
	Prácticas Externas	<i>Completar la formación de los alumnos en un contexto profesional</i>
	Tutorías	<i>Atención personalizada a los estudiantes</i>
	Trabajo en grupo	<i>Hacer que los estudiantes aprendan entre ellos</i>
	Trabajo autónomo	<i>Desarrollar la capacidad de autoaprendizaje</i>



Tabla 1. Métodos de enseñanza

MÉTODOS DE ENSEÑANZA		
	Método	Finalidad
	Método Expositivo/Lección Magistral	Transmitir conocimientos y activar procesos cognitivos en el estudiante
	Estudio de Casos	Adquisición de aprendizajes mediante el análisis de casos reales o simulados
	Resolución de Ejercicios y Problemas	Ejercitar, ensayar y poner en práctica los conocimientos previos
	Aprendizaje Basado en Problemas (ABP)	Desarrollar aprendizajes activos a través de la resolución de problemas
	Aprendizaje orientado a Proyectos	Realización de un proyecto para la resolución de un problema, aplicando habilidades y conocimientos adquiridos
	Aprendizaje Cooperativo	Desarrollar aprendizajes activos y significativos de forma cooperativa
	Contrato de Aprendizaje	Desarrollar el aprendizaje autónomo

Se conoce como método expositivo "la presentación de un tema lógicamente estructurado con la finalidad de facilitar información organizada siguiendo criterios adecuados a la finalidad pretendida". Esta metodología -también conocida como lección (lecture)- se centra fundamentalmente en la exposición verbal por parte del profesor de los contenidos sobre la materia objeto de estudio. El término "lección magistral" se suele utilizar para denominar un tipo específico de lección impartida por un profesor en ocasiones especiales.

Análisis intensivo y completo de un hecho, problema o suceso real con la finalidad de conocerlo, interpretarlo, resolverlo, generar hipótesis, contrastar datos, reflexionar, completar conocimientos, diagnosticarlo y, en ocasiones, entrenarse en los posibles procedimientos alternativos de solución.

Situaciones en las que se solicita a los estudiantes que desarrollen las soluciones adecuadas o correctas mediante la ejercitación de rutinas, la aplicación de fórmulas o algoritmos, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados. Se suele utilizar como complemento de la lección magistral.

Método de enseñanza-aprendizaje cuyo punto de partida es un problema que, diseñado por el profesor, el estudiante ha de resolver para desarrollar determinadas competencias previamente definidas.

Método de enseñanza-aprendizaje en el que los estudiantes llevan a cabo la realización de un proyecto en un tiempo determinado para resolver un problema o abordar una tarea mediante la planificación, diseño y realización de una serie de actividades, y todo ello a partir del desarrollo y aplicación de aprendizajes adquiridos y del uso efectivo de recursos.

Enfoque interactivo de organización del trabajo en el aula en el cual los alumnos son responsables de su aprendizaje y del de sus compañeros en una estrategia de corresponsabilidad para alcanzar metas e incentivos grupales.

Es tanto un método, a utilizar entre otros, como un enfoque global de la enseñanza, una filosofía.

Un acuerdo establecido entre el profesor y el estudiante para la consecución de unos aprendizajes a través de una propuesta de trabajo autónomo, con una supervisión por parte del profesor y durante un período determinado. En el contrato de aprendizaje es básico un acuerdo formalizado, una relación de contraestación recíproca, una implicación personal y un marco temporal de ejecución.





	<p>Durante una clase de teoría o lección magistral, el profesor realiza una exposición verbal de los contenidos sobre la materia objeto de estudio, mediante la cual suministra a los alumnos información esencial y organizada procedente de diversas fuentes con unos objetivos específicos predefinidos (motivar al alumno, exponer los contenidos sobre un tema, explicar conocimientos, efectuar demostraciones teóricas, presentar experiencias, etc.) pudiendo utilizar para ello, además de la exposición oral, otros recursos didácticos (audiovisuales, documentos, etc).</p>
	<p>Este método de enseñanza se utiliza como complemento de la clase de teoría (lección magistral) y se basa en solicitar a los estudiantes que desarrollen las soluciones adecuadas o correctas mediante la ejercitación de rutinas, la aplicación de fórmulas o algoritmos, la aplicación de procedimientos de transformación de la información disponible y la interpretación de los resultados.</p> <p>La intención principal es la de aplicar lo ya aprendido para favorecer la comprensión tanto de la importancia como del contenido de un nuevo tema, afianzar conocimientos y estrategias y su aplicación en las situaciones prácticas que se planteen.</p>
	<p>Clases que se imparten en aula informática con acceso a los recursos computacionales que se necesitan para el desarrollo de la materia. El alumno trabaja individualmente o en grupos muy reducidos (2-3 estudiantes) en la implementación y aplicación de un algoritmo bajo la supervisión del profesor.</p>



8. Recursos didácticos

RECURSOS DIDÁCTICOS	
BIBLIOGRAFÍA	“Matrix Computation”, Jennings, McKeown, Wiley 1992
	“Lapack User's Guide”, Anderson et al. Society for Industrial and Applied Mathematics (SIAM) 1992
	“Parallel Programming in C with OpenMP and MPI”, M. Quinn, McGraw-Hill 2003
	“Fast Algorithms for Digital Signal Processing”, Blahut, Addison-Wesley, Reading MA, 1985
	“High Performance Cluster Computing”, Buyya, Prentice-Hall 1999
RECURSOS WEB	Página web de la asignatura (http://)
	http://www.netlib.org
	http://www.openmp.org
	http://www.mpi-forum.org/docs/docs.html http://www.mcs.anl.gov/research/projects/mpich2/
EQUIPAMIENTO	Laboratorio
	Aula XXXX
	Sala de trabajo en grupo



Cronograma de trabajo de la asignatura

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	
Semanas 1 -- 2	<ul style="list-style-type: none"> Tema 1: Algorítmica numérica. (3h/semana) 	<ul style="list-style-type: none"> Tema 1: Algorítmica numérica. (3h/semana) 	<ul style="list-style-type: none"> Estudio de algoritmos numéricos. (17 h) 	<ul style="list-style-type: none"> * Trabajos de preparación para la definición y defensa del primer proyecto.(5 h) 	<ul style="list-style-type: none"> Prime Defini del pr
Semanas 3 -- 6	<ul style="list-style-type: none"> Tema 2: Programación y Librerías Numéricas. (3h/semana) 	<ul style="list-style-type: none"> Tema 2: Programación y Librerías Numéricas. (3h/semana) 	<ul style="list-style-type: none"> Estudio de técnicas de optimización y programación paralela (6 h) 	<ul style="list-style-type: none"> Implementación práctica del primer proyecto.(19h) Preparación defensa solución primer proyecto.(4h) Preparación memoria escrita. (6h) 	<ul style="list-style-type: none"> Pro de imp Pro esc
Semanas 7-8	<ul style="list-style-type: none"> Tema 3: Arquitecturas y Sistemas. (3h/semana) 	<ul style="list-style-type: none"> Tema 3: Arquitecturas y Sistemas. (3h/semana) 	<ul style="list-style-type: none"> Estudio de arquitectura y redes de sistemas de alto rendimiento. (12 h) 	<ul style="list-style-type: none"> * Trabajos de preparación para la definición y defensa del primer proyecto.(6 h) * Implementación práctica segundo proyecto. (19 h) * Preparación defensa solución primer proyecto. (4h) 	<ul style="list-style-type: none"> Pro y d pro Pro de imp
Semana Exámenes				<ul style="list-style-type: none"> Preparación memoria escrita. (6h) 	<ul style="list-style-type: none"> pro esc

Nota: Para cada actividad se especifica la dedicación en horas que implica para el alumno.

Horas presenciales en aula: 6 x 8 h/semana = 48

Horas de trabajo individual/grupo = 104.

Total : 152 horas (6 créditos)