



Técnicas de Computación Científica

Guía de Aprendizaje – Información al estudiante

1. Datos Descriptivos

Asignatura	Técnicas de Computación Científica
Materia	Ingeniería
Departamento responsable	Lenguajes y Sistemas Informáticos en Ingeniería del Software
Créditos ECTS	6
Carácter	Optativa
Titulación	Grado en Matemáticas más Informática por la Universidad Politécnica de Madrid
Curso	4º
Especialidad	No aplica

Curso académico	2013-2014
Semestre en que se imparte	Sept.-Enero (puede cambiar según planificación docente)
Semestre principal	Sept.-Enero
Idioma en que se imparte	Español
Página Web	



2. Profesorado

NOMBRE Y APELLIDO	DESPACHO	Correo electrónico
Vicente Martín Ayuso	5210	vicente@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">• Conocimientos básicos de algorítmica y arquitectura de ordenadores.



4. Objetivos de Aprendizaje

COMPETENCIAS ASIGNADAS A LA ASIGNATURA Y SU NIVEL DE ADQUISICIÓN		
Código	Competencia	Nivel
CE	Capacidad para usar consistentemente los recursos computacionales.	3
CE-2	Formalización y la especificación de problemas reales cuya solución requiere el uso de la informática	4
CE-3	Capacidad de elegir y usar los métodos analíticos y de modelización relevantes	3
CE-12	Adquisición de conocimientos avanzados en el campo de ciencias de la computación	4
CE-53	Capacidad de trabajar de forma efectiva como individuo y como miembro de un equipo	3
CE-15	Capacidad de leer y comprender así como de catalogar y clasificar científicamente publicaciones dentro de su campo de estudio.	3

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



POLITÉCNICA

UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
Campus de Montegancedo
Boadilla del Monte. 28660 Madrid

RESULTADOS DE APRENDIZAJE DE LA ASIGNATURA			
Código	Resultado de aprendizaje	Competencias asociadas	Nivel de adquisición
RA1	Los estudiantes usan eficientemente los recursos computacionales	CU , CE3, CE53-54	4
RA2	Los estudiantes comprenden y evalúan las distintas técnicas que influyen en el uso eficiente de un ordenador, secuencial y paralelo.	CU,CE2-4,C54	
RA3	Los estudiantes son capaces de aplicar técnicas de optimización y paralelización para resolver problemas reales.	CU, CE2-3, CE 53-54	



5. Sistema de evaluación de la asignatura

INDICADORES DE LOGRO		
Ref	Indicador	Relaciona-do con RA
I1	Diseñar e implementar programas que usen de manera eficiente los recursos computacionales de ordenadores secuenciales.	RA1, RA3
I2	Conocer las implicaciones de rendimiento que conlleva la implementación en máquina de los algoritmos.	RA1, RA3
I3	Usar ordenadores de alto rendimiento para ejecutar aplicaciones en ciencia e ingeniería.	RA2
I4	Diseñar e implementar una aplicación paralela para resolver un problema real.	RA2, RA3

EVALUACION SUMATIVA			
Breve descripción de las actividades evaluables	Momento	Lugar	Peso en la calif.
Proyecto 1: Problemas secuenciales.	Durante el curso (ver CRONOGRAMA) Fecha definitiva a confirmar en clase	Entrega por medios electrónicos.	40% Se precisa superar un 3.0
Proyecto 2: Problemas paralelos y aplicación práctica.	Durante el curso (ver CRONOGRAMA) fecha límite exacta a confirmar en clase)	Entrega por medios electrónicos.	60% Se precisa superar un 3.0
			Total: 100%



CRITERIOS DE CALIFICACIÓN

El método de evaluación estará basado en la ejecución de dos proyectos durante el curso. Uno de ellos estará centrado en arquitecturas secuenciales y otro en arquitecturas paralelas y aplicaciones prácticas. Los proyectos tienen un peso del 40% el primero y del 60% el segundo. Previamente a cada proyecto habrá una fase de definición del mismo en el que se discutirá el trabajo a realizar. Los proyectos se entregarán en formato electrónico, incluyendo los códigos desarrollados si es el caso. Si se considera necesario, el proyecto puede ser requerido para su defensa presencial.

La convocatoria extraordinaria de Julio y la evaluación mediante solo prueba final, consistirá en un examen de teoría y otro de práctica en aula informática donde tendrá que resolver problemas escribiendo programas o modificando otros que le serán presentados. Ambas pruebas tienen el mismo peso en la nota final. Es necesaria una nota superior a 3 puntos sobre 10 en ambos exámenes y una nota media superior a 5 puntos sobre 10 para aprobar la asignatura.

El alumno que desee seguir el sistema de evaluación mediante sólo prueba final, deberá solicitarlo mediante escrito dirigido al coordinador de la asignatura y a través del Registro de la Secretaría de Alumnos en el plazo de quince días a contar desde el inicio de la actividad docente de la asignatura. El modelo de solicitud se encuentra disponible en Secretaría de Alumnos.



POLITÉCNICA

UNIVERSIDAD POLITÉCNICA DE MADRID
FACULTAD DE INFORMÁTICA
Campus de Montegancedo
Boadilla del Monte. 28660 Madrid

CRITERIOS DE CALIFICACIÓN



6. Contenidos y Actividades de Aprendizaje

CONTENIDOS ESPECÍFICOS		
Bloque / Tema / Capítulo	Apartado	Indicadores Relacionados
Tema 1: optimización en arquitecturas secuenciales.	<ul style="list-style-type: none">• Overview de arquitecturas y subsistemas de memoria.• Optimización del subsistema de memoria.• Optimización de la CPU	I1,I2
Tema 2: Introducción al paralelismo y arquitecturas paralelas	Aarquitecturas paralelas. Evolución de las mismas, expectativas y herramientas.	I2
Tema 3: Programación paralela.	Principales paradigmas de programación paralela y programación con aceleradores: HPF, OpenMP, OpenACC, MPI y UPC. Benchmarking.	I1,I2,I3
Tema 4: Green computing	Eficiencia energética en instalaciones y ordenadores de alto rendimiento.	I1,I2
Tema 5: Programación de coprocesadores	Programación de los coprocesadores más habituales en computación de alto rendimiento: GPUs e Intel MICs	I1, I2
Tema 6: Aplicaciones en Computación Científica	Introducción a algunas de las áreas de aplicación típicas en computación de alto rendimiento: Visualización, Dinámica de fluidos computacional, bioinformática...	I1,I2,I3

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 8. 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 contraprestación 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	<ul style="list-style-type: none"> • High Performance Cluster Computing. R. Buyya. Ed. Prentice Hall. 1999 • Cluster Computing White Paper. M. Baker, et al. 2001. • Using MPI, Portable Parallel Programming with the Message Passing Interface. W. Gropp, E. Lusk, A. Skjellum. Ed. MIT Press. 1999 • Message Passing Interface Forum http://www.mpi-forum.org/ • The High Performance Fortran Handbook. Scientific and Engineering Computation Series C.H. Koelbel et al. The MIT Press, 1994. • UPC: Distributed Shared Memory Programming. T. El-Ghazawi et al. Wiley Series on Parallel and Distributed Computing, Wiley Interscience. 2005 • Berkeley Unified Parallel C (UPC) Project. http://upc.lbl.gov • Parallel Programming in OpenMP. R. Chandra et al. Ed. Morgan Kaufmann, 2001. • OpenMP Forum http://openmp.org/ • OpenACC standard http://www.openacc-standard.org/ • The Green Grid: http://www.thegreengrid.org
RECURSOS WEB	<p>Página web de la asignatura http://www.fi.upm.es/~vicente/tcc/tcc.html</p>
	<p>Para cada tema se aportará material para guiar y facilitar el estudio que estará disponible en la pagina web. Típicamente por cada tema habrá:</p> <ol style="list-style-type: none"> 1) Transparencias con el desarrollo del tema. 2) Guías de practicas Ejercicios de cara a preparar las pruebas de laboratorio. 3) Bibliografía extendida en página web

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	Actividades de Evaluación	Otros
Semanas 1	<ul style="list-style-type: none"> Introducción. Arquitecturas secuenciales. (2h) 	<ul style="list-style-type: none"> INTRODUCCIÓN (1h) 	<ul style="list-style-type: none"> 			<ul style="list-style-type: none">
Semanas 2 -- 3	<ul style="list-style-type: none"> Perfilado. Uso eficiente de la jerarquía de memoria. (6 horas = 4 Teo + 2 Lab) Seminarios: Aplicaciones (6 horas) 	<ul style="list-style-type: none"> Clase Lab (2h) 	<ul style="list-style-type: none"> 8 horas (conceptos/pract) 	<ul style="list-style-type: none"> 4 horas (proyecto) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none">
Semanas 4 -- 7	<ul style="list-style-type: none"> Optimización en la CPU librerías optimizadas, Benchmarking(12 horas= 6 Teo+6 Lab) Seminarios: Aplicaciones (12 horas) 	<ul style="list-style-type: none"> Clase LAB (6 h) 	<ul style="list-style-type: none"> 16 horas (conceptos/pract) 	<ul style="list-style-type: none"> 4 horas (proyecto) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none">
Semanas 8 -- 9	<ul style="list-style-type: none"> Paralelismo y arquitecturas paralelas (6 horas = 4 Teo + 2 Lab) Seminario de Visualización científica (6 horas, 3 Teo + 3 Lab) 	<ul style="list-style-type: none"> Clase LAB (5 h) 	<ul style="list-style-type: none"> 8 horas (conceptos/pract) 	<ul style="list-style-type: none"> 4 horas (proyecto) 	<ul style="list-style-type: none"> Entrega proyecto 1 	<ul style="list-style-type: none">
Semanas 10 --12	<ul style="list-style-type: none"> Programación paralela openMp, HPF (9 horas=4 Teo+5 Lab) Seminario de programación de coprocesadores (GPU+MIC) (9 horas=4 Teo+5 Lab) 	<ul style="list-style-type: none"> Clase LAB (10 h) 	<ul style="list-style-type: none"> 12 horas (conceptos/pract) 	<ul style="list-style-type: none"> 4 horas (proyecto) 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none">

Semana	Actividades en Aula	Actividades en Laboratorio	Trabajo Individual	Trabajo en Grupo	Actividades de Evaluación	Otros
	•					
Semanas 13--15	<ul style="list-style-type: none"> • Programación paralela . UPC, MPI (9 horas=4 Teo+5 Lab) • Seminario de programación de coprocesadores (GPU+MIC) (9 horas=4 Teo+5 Lab) 	<ul style="list-style-type: none"> • Clase LAB (10 h) 	<ul style="list-style-type: none"> • 12 horas (conceptos/pract) 	<ul style="list-style-type: none"> • 4 horas (proyecto) 	•	•
Semana Exámenes	EXAMEN FINAL TEORIA Solo para alumnos que no superen los proyectos	EXAMEN FINAL LABORATORIO Solo para alumnos que no superen los proyectos	Se supone un seguimiento previo asignatura. El examen final de LABORATORIO no es obligatorio y puede aprobarse por curso.		<ul style="list-style-type: none"> • Entrega proyecto 2 • EXAMEN FINAL de PROBLEMAS y LABORATORIO (opcional) 	•

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

COMENTARIOS al CRONOGRAMA ANTERIOR:

1. Las horas reservadas en nuestro horario para ACTIVIDADES en AULA suman 84, correspondientes a un horario de 6 h/semana x 15 semanas. En el cronograma anterior, las horas listadas suman 82 horas (dejando margen para fiestas, imponderables).
2. El cronograma distribuye los seminarios de manera razonablemente homogénea y acorde con los conocimientos impartidos en cada tema. Los temas específicos de los seminarios de aplicación no están todavía definidos. Variarán de un año a otro y su situación en el cronograma podría sufrir variaciones.
3. Las horas de LABORATORIO (prácticas) programadas suman un máximo de 34 horas, cerca de la mitad de las horas ajustadas a un horario promedio aproximado de 2,5 horas/semana que se distribuirán, tal y como figura en el cronograma y dentro de la distribución de seminarios, de manera lo más uniforme posible conforme a los horarios que en su día asigne Jefatura de Estudios.

4. El computo de horas de trabajo del alumno fuera del aula es de 76 horas, que sumadas a las 84 en aula (notese que hay mucha carga de trabajo de practicas en horario lectivo) dan las 160 horas correspondientes a 6 creditos.

