



ESCUELA POLITECNICA SUPERIOR

Guía Docente

Curso 2009-2010

Titulación

Ingeniería Técnica Industrial - Electrónica Ind.

DATOS DE LA ASIGNATURA*

* Asignatura en experiencia piloto de implantación del sistema de créditos ECTS

Nombre:			
Informática			
Denominación en inglés¹:			
Computer science			
Código:	Año del Plan de Estudios:	Tipo:	
310099028	Publicación BOE: 20-05-1999	<input type="checkbox"/> Troncal <input type="checkbox"/> Obligatoria <input checked="" type="checkbox"/> Optativa	
Créditos:			
	Totales:	Teóricos:	Prácticos:
Créditos L.R.U.	6,00	3,00	3,00
Créditos E.C.T.S.	4,8	2,4	2,4
Departamento:			
Tecnologías de la Información			
Area de Conocimiento:			
Lenguaje y Sistemas Informáticos			
Curso:	Cuatrimestre:	Ciclo:	
Tercero	1º Cuatrimestre	Primero	
Web de la asignatura:			
Plataforma Moodle (http://www.uhu.es/sevirtual/)			

¹Para su inclusión en el Complemento Europeo al Título

DATOS DE LOS PROFESORES

Nombre:	e-mail:	Teléfono:	Despacho:
María del Pilar Polo Almohano	polo@dti.uhu.es	959 217386	Edificio. TorreUmbría Dpcho nº 56

DATOS ESPECÍFICOS DE LA ASIGNATURA

1.1. Descriptores de la asignatura:

Ficheros. Registros. Programación de periféricos. Estructuras dinámicas de datos. Pilas. Colas. Árboles. Recursividad

1.2. Descriptores de la asignatura (en inglés)²:

Files. Records. Programming of peripheral. Dynamical data structures. Stacks. Tails. Trees. Recurring (Recursion)

²Para su inclusión en el Complemento Europeo al Título

2. Situación de la asignatura.

2.1. Prerrequisitos:

- Tener un nivel medio/bajo en comprensión de texto en inglés.
- Posibilidad de tener ordenador en casa.

2.2. Contexto dentro de la titulación:

Por sus contenidos, de acuerdo con los descriptores del BOE, y dado el marcado carácter instrumental de la materia y que la mayor parte de las asignaturas de Ingeniería necesitan métodos de cálculo susceptibles de ser realizados con ayuda de ordenador, observamos que podrían ser todas, prácticamente, las materias a las que la asignatura optativa *Informática* apoyará desde sus contenidos, como herramienta indispensable para el ingeniero. Esta asignatura reforzará y ampliará los conocimientos adquiridos por los alumnos en la materia troncal *Fundamentos de Informática*, dotándolos de mayor capacidad a la hora de resolver problemas y de implementarlos en un lenguaje de programación.

2.3. Recomendaciones:

- Haber cursado y aprobado la asignatura *Fundamentos de Informática*.
- El alumno deberá estudiar la asignatura consultando la bibliografía sugerida por los profesores y asistir con regularidad a las tutorías que estos ofertan.
- Debido al marcado carácter práctico de la asignatura, se recomienda al alumnado realizar la mayor parte de los ejercicios propuestos en las relaciones de problemas que se les proporcionen.

3. Competencias a adquirir por los estudiantes.

3.1. Competencias transversales o genéricas.

3.1.1. Competencias instrumentales:

<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Capacidad de análisis y síntesis.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Capacidad de organización y planificación.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Comunicación oral y escrita en lengua nativa.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input checked="" type="checkbox"/> Bajo	Conocimiento de una lengua extranjera.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Capacidad de gestión de la información.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Resolución de problemas.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Toma de decisiones.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Conocimientos generales básicos.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Conocimientos básicos de la profesión.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Conocimientos de informática.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.

3.1.2. Competencias personales:

<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input checked="" type="checkbox"/> Bajo	Trabajo en equipo.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Trabajo en un equipo de carácter interdisciplinar.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Trabajo en un contexto internacional.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Habilidades en las relaciones interpersonales.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input checked="" type="checkbox"/> Bajo	Capacidad para comunicarse con expertos de otras áreas.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Reconocimiento a la diversidad y la multiculturalidad.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Razonamiento crítico.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Compromiso ético.

<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.
3.1.3. Competencias sistémicas:			
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Aprendizaje autónomo.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Adaptación a nuevas situaciones.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Capacidad de aplicar los conocimientos en la práctica.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Habilidad para trabajar de forma autónoma.
<input checked="" type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Creatividad.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Liderazgo.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Conocimiento de otras culturas y costumbres.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Iniciativa y espíritu emprendedor.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Motivación por la calidad.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Sensibilidad hacia temas medioambientales.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Diseño y gestión de proyectos.
<input type="checkbox"/> Alto	<input checked="" type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Motivación de logro.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.
<input type="checkbox"/> Alto	<input type="checkbox"/> Medio	<input type="checkbox"/> Bajo	Otras: Especificar.
3.2. Competencias específicas.			
3.2.1. Competencias cognitivas (saber):			
Nuevas Tecnologías (E3).			
Conocimientos de Informática (E18).			
3.2.2. Competencias procedimentales e instrumentales (saber hacer):			
Métodos de Diseño (E7): en la materia <i>Informática</i> se hace especial hincapié en las metodologías de diseño de programas, que pueden en muchos casos, extrapolarse a otras materias.			
Planificación y Programación (E21): la realización de un algoritmo requiere una capacidad de organización y de planificación previa, así como una capacidad de abstracción, vital para su posterior desarrollo.			
3.2.2. Competencias actitudinales (ser):			
Aprendizaje Autónomo (G15): los métodos de aprendizaje de la materia exigen la necesidad de incrementar la capacidad autónoma de razonamiento y análisis, proporcionando además la capacidad posterior al alumno para continuar el aprendizaje LLL (Long Life Learning), en ésta o en otras materias.			
Creatividad (G17): la resolución de nuevos problemas mediante la programación, requiere un esfuerzo creativo de diseño. La programación, como acto final de la resolución de un problema, es en sí, un acto creativo.			

4. Objetivos:
Afianzar los conocimientos adquiridos en la asignatura <i>Fundamentos de Informática</i> y dotar al alumno de mayor capacidad para la resolución de problemas mediante la programación, ampliando sus conocimientos con los siguientes conceptos:
ESTRUCTURAS
<ul style="list-style-type: none"> Se reforzará el concepto de estructura (ya estudiado en <i>Fundamentos de Informática</i>). Se desarrollarán prácticas con las que el alumno comprenda la importancia y utilidad de este tipo de dato.
FICHEROS
<ul style="list-style-type: none"> Hasta el momento de estudiar ficheros (archivos) los programas que se habían diseñado disponían sólo de la posibilidad de almacenar información en memoria principal, lo que implicaba que al dejar de ejecutarse, sus datos desaparecían, de forma que en cada ejecución debían reintroducirse los datos con los que trabajaban. El volumen de datos que los programas podían manejar estaba condicionado por la disponibilidad de memoria principal. El concepto de fichero (archivo) acabará con los problemas anteriores, pues nos permitirá leer y almacenar información en dispositivos de almacenamiento no volátiles y de mayor capacidad, pudiendo de este modo, cuando sea necesario, salvar de forma permanente la información generada en nuestros programas para su almacenamiento o posterior tratamiento, evitando que dicha información se pierda al terminar la ejecución del programa que lo genera.
RECURSIVIDAD

- Es una herramienta fundamental en el tratamiento de problemas, especialmente matemáticos y científicos, pues muchos de ellos tienen naturaleza recursiva.
- El concepto de recursividad, su tratamiento y el estudio de algoritmos de resolución son de gran importancia en la formación de todo programador.

TIPOS ABSTRACTOS DE DATOS (TAD)

- El concepto de TAD es la base fundamental de la programación moderna.
- Se estudiarán los TAD Pila, Cola y Árbol

5. Metodología (en horas de trabajo del estudiante):

	Primer Cuatrimestre	Segundo Cuatrimestre
	Presenciales	
Clases de teoría	17,0	0,0
Clases de problemas	13,0	0,0
Clases prácticas	26,0	0,0
Actividades académicas dirigidas	4,0	0,0
	Exámenes	
	6,0	0,0
	No presenciales	
Estudio de clases teóricas (factor de trabajo: 0,80)	13,6	0,0
Estudio de clases de problemas y prácticas (factor de trabajo: 0,95)	37,0	0,0
Preparación de actividades académicamente dirigidas y otras actividades	11,5	0,0
Total:	128,1	0,0
Trabajo total del estudiante: 128,1 horas.		
Horas presenciales:	60,0	Horas no presenciales: 62,1
		Exámenes: 6,0

6. Técnicas docentes.

6.1. Técnicas docentes utilizadas:

- Sesiones académicas de teoría
- Sesiones académicas de problemas
- Sesiones prácticas en laboratorio
- Seminarios, exposiciones y debates
- Trabajo en grupos reducidos
- Resolución y entrega de problemas/prácticas
- Realización de pruebas parciales evaluables
- Otras: Trabajos o Actividades Académicamente Dirigidas
- Otras: Especificar

6.2. Desarrollo y justificación:

Sesiones Académicas de Teoría:

En las clases magistrales se impartirá, de una manera clara y concisa, la base teórica de la asignatura y se expondrán ejemplos aclaratorios de la misma. Las sesiones serán de dos horas y se irán intercalando con las sesiones de problemas a lo largo del curso, de manera que una vez finalizado un tema teórico con sus correspondientes sesiones académicas de teoría, se impartirán sesiones de problemas.

La metodología usada para impartir la teoría y los ejemplos aclaratorios será la exposición mediante transparencias y uso de pizarra.

En la plataforma Moodle de la asignatura se encontrarán las transparencias y otros materiales de referencia necesarios para el seguimiento de las sesiones.

Sesiones de Problemas:

Consisten en la realización de problemas relacionados con los conceptos y métodos operativos de la asignatura. El profesor explicará uno o varios problemas tipo. En estas sesiones se fomentará la participación del alumnado en la resolución de los problemas planteados.

Para la convocatoria de Febrero, se **realizarán a lo largo del trimestre un total de 2 pruebas puntuables consistentes en la entrega al profesor de la solución a un ejercicio propuesto.**

En la plataforma Moodle de la asignatura se encontrará un compendio de problemas para resolver, y se irán indicando aquellos que van siendo resueltos en las sesiones. El resto de problemas deberán ser resueltos por los alumnos de forma voluntaria y las soluciones propuestas por ellos podrán ser comprobadas haciendo uso de las horas de tutorías y de atención al alumno del profesor.

Prácticas de Laboratorio:

Consisten en el diseño e implementación de programas escritos en C. Los alumnos dispondrán con antelación la relación de problemas a resolver y la metodología de trabajo. El trabajo se realizará de forma individual siendo las sesiones de dos horas.

La **asistencia** a las sesiones de laboratorio es **obligatoria**.

Trabajos o Actividades Académicamente Dirigidas.

Se realizarán **dos pruebas de progreso**, a lo largo del curso, con el fin de llevar una evaluación continua, a la par que cada alumno pueda realimentarse de sus propios resultados midiendo así la evolución de su aprendizaje.

Las pruebas consistirán en un problema práctico a resolver mediante la construcción de un programa en C en el laboratorio.

7. Bloques temáticos:

I. FUNDAMENTOS DE PROGRAMACIÓN

1. REPASO DE CONCEPTOS DE INFORMÁTICA BÁSICA Y PROGRAMACIÓN ESTRUCTURADA

II. PROGRAMACIÓN AVANZADA

1. CONCEPTO DE ESTRUCTURA
2. FICHEROS (ARCHIVOS)
3. RECURSIVIDAD
4. TIPOS ABSTRACTOS DE DATOS (T.A.D.)

8. Temario desarrollado:

Bloque Temático 1: FUNDAMENTOS DE PROGRAMACIÓN

TEMA 1: REPASO DE CONCEPTOS DE INFORMÁTICA BÁSICA Y PROGRAMACIÓN ESTRUCTURADA

- 1.1. Concepto de ordenador.
 - 1.1.1. Componentes de un ordenador.
 - 1.1.1.1. Unidades de Almacenamiento.
 - 1.1.1.2. Unidad Central de Proceso (C.P.U.)
 - 1.1.1.3. Dispositivos de E/S (Periféricos).
 - 1.1.1.4. Buses.
 - 1.2. Concepto de S.O.
 - 1.3. Fundamentos de Programación. Repaso del lenguaje C.
 - 1.3.1. Conceptos básicos de algorítmica.
 - 1.3.2. Proceso de compilación.
 - 1.3.3. Elementos básicos de programación.
 - 1.3.3.1. Estructura general de un programa en C.
 - 1.3.3.2. Tipos de Datos.
 - 1.3.3.3. Variables.

- 1.3.3.4. Constantes.
- 1.3.3.5. Operadores.
- 1.3.3.6. Operaciones básicas de E/S.
- 1.3.3.7. La Librería de C.
- 1.3.4. Control de flujo del programa.
- 1.3.4.1. Estructuras de Selección.
- 1.3.4.2. Estructuras de Repetición.
- 1.3.5. Diseño Descendente: Subprogramas.
- 1.3.5.1. Subprogramas: Procedimientos y Funciones.
- 1.3.5.2. Parámetros y Variables Locales. Variables Globales.
- 1.3.5.3. Parámetros por Valor.
- 1.3.5.4. Parámetros por Referencia.
- 1.3.5.5. Prototipos.
- 1.3.5.6. Módulos.
- 1.3.5.6.1. Ámbito de los subprogramas.
- 1.3.5.6.2. Proyectos.
- 1.3.5.7. Sección de *#includes*.
- 1.3.6. Estructuras de datos complejas: Matrices.
- 1.3.6.1. Inicialización de Vectores.
- 1.3.6.2. Cadenas (string).
- 1.3.6.3. Paso de arrays a subprogramas.
- 1.4. Ejercicios de repaso.

Bloque Temático 2: PROGRAMACIÓN AVANZADA

TEMA 2: CONCEPTO DE ESTRUCTURA

- 2.1. Concepto, declaración y definición.
- 2.2. Acceso a los campos de una Estructura.
- 2.3. Estructuras anidadas.
- 2.4. Arrays de estructuras.
- 2.5. Paso de Estructuras a subprogramas.
- 2.5.1. Punteros a Estructuras.
- 2.5.2. Paso de Elementos de Estructuras (campos) a subprogramas
- 2.6. Typedef.
- 2.7. Ejercicio Práctico.

TEMA 3: FICHEROS (ARCHIVOS)

- 3.1. Ficheros en C.
- 3.1.1. Tipos de archivos.
- 3.2. Operaciones con ficheros.
- 3.2.1. Archivos Lógicos y Archivos Físicos.
- 3.2.2. Declaración de Archivos Lógicos.
- 3.2.3. Creación y Apertura de Ficheros.
- 3.2.4. Cierre de Ficheros.
- 3.2.5. Lectura y Escritura de Ficheros.
- 3.2.6. Acceso Directo.
- 3.2.7. Operaciones Auxiliares.
- 3.3. Ejercicios.

TEMA 4: RECURSIVIDAD

- 4.1. Introducción a la recursividad. Concepto de algoritmo recursivo.
- 4.1.1. Ejemplos básicos.
- 4.1.2. Seguimiento de una función recursiva.
- 4.1.3. Implementación.
- 4.2. Recursividad directa e indirecta.
- 4.3. Recursividad simple (lineal) y múltiple.
- 4.4. Métodos para la resolución de problemas recursivos.
- 4.4.1. Divide y vencerás.
- 4.4.2. Backtracking (retroceso)
- 4.5. Ejercicios.

TEMA 5: TIPOS ABSTRACTOS DE DATOS (T.A.D.)

- 5.1. Concepto de TAD.
- 5.2. Concepto de Estructura de Datos Dinámica.
- 5.3. TAD Pila

- 5.3.1. Definición y Operaciones.
- 5.3.2. Implementaciones.
- 5.4. TAD Cola
 - 5.4.1. Definición y Operaciones.
 - 5.4.2. Implementaciones.
- 5.4. TAD Árbol Binario
 - 5.4.1. Definición y Operaciones.
 - 5.4.2. Implementaciones.
- 5.5 Ejercicios.

9. Bibliografía.

9.1. Bibliografía general:

- Fernández, Fco. Javier; Márquez, Antonio A.; Polo, M^a Pilar. "Fundamentos de Informática: Introducción a la Programación en C". Servicio de Publicaciones Universidad de Huelva. 2005.
- Delannoy, Claude. "El libro del C. Primer lenguaje". Ediciones Gestión 2000, S.A.
- Joyanes, L; Sánchez, L; Fernández, M; Zahonero, I. "Estructuras de Datos en C". Ed. McGraw-Hill, 2005
- Joyanes, L; Zahonero, I. "Programación en C: Metodología, estructura de datos y objetos". Ed. McGraw-Hill. 2001

9.2. Bibliografía específica:

- Prieto, A; Lloris, A; Torres, J.C. "Introducción a la Informática". Ed. McGraw-Hill
- Joyanes, L. "Fundamentos de Programación. Algoritmos y Estructuras de Datos". Ed. McGraw-Hill
- Anasagasti, M. "Fundamentos de Computadores", Ed. Paraninfo.
- Aho, Alfred V.; Hopcroft, John E.; Ullman, Jeffrey D. "Estructuras de datos y algoritmos". Ed. Addison-Wesley Iberoamericana

10. Técnicas de evaluación.

10.1. Técnicas de evaluación utilizadas:

- Examen teórico-práctico
- Trabajos desarrollados durante el curso
- Participación activa en las sesiones académicas
- Controles periódicos de adquisición de conocimientos
- Examen práctico en aula de informática
- Otras: Controles periódicos de adquisición de conocimientos en el aula de informática.
- Otras: Resolución de problemas.

10.2. Criterios de evaluación y calificación:

Se resumen en esta tabla:

Aspecto	Criterio	Instrumento		Peso Conv. Febrero	Peso Restantes Conv.
Conocimientos teóricos	Exposición escrita de conocimientos y su aplicación a distintas situaciones	Examen Teórico Escrito		70%	80%
Conocimientos prácticos	Resolución de Problemas	Problema 1 (aula teoría)	5%	10%	0%
		Problema 2 (aula teoría)	5%		
	Capacidad de implementar programas en C	Control 1 (aula informática)	10%	20%	20%
		Control 2 (aula informática)	10%		

La nota en acta del alumno, para la convocatoria de Febrero será:

A	B	C	
Nota _(Teórico)	Nota _(Resolución de Problemas)	Nota _(Controles Prácticos)	Nota en Acta
≥ 4 (sobre 10)	$\sum_{Pr\ oblema=1}^2 Nota_{Pr\ oblema}$	$\sum_{Control=1}^2 Nota_{Control}$	$(A*0.7) + B + C$
< 4 (sobre 10)	$\sum_{Pr\ oblema=1}^2 Nota_{Pr\ oblema}$	$\sum_{Control=1}^2 Nota_{Control}$	A

La nota en acta del alumno, para las convocatorias de Septiembre, Noviembre o Diciembre será:

A	C	
Nota _(Teórico)	Nota _(Examen Práctico)	Nota en Acta
≥ 4 (sobre 10)	Nota	$(A*0.8) + C$
< 4 (sobre 10)	Nota	A

Criterio para guardar las notas para convocatorias posteriores:

- La nota del **examen teórico** se guarda hasta la convocatoria de Diciembre, siempre que se obtenga una nota igual o superior a 5 (sobre 10).
- La nota de la **resolución de problemas NO se guarda** para posteriores convocatorias, lo que significa que el examen teórico de Septiembre, Noviembre y Diciembre, tendrá un peso del 80% de la Nota en Acta.
- La nota de los **controles prácticos** se guarda hasta la convocatoria de Diciembre, siempre se que haya obtenido una nota igual o superior a 1 (sobre 2).
- No obstante cualquier alumno podrá presentarse en septiembre para ser reevaluado, tanto en teoría como en prácticas, si así lo desea (y siempre que se lo haya solicitado al profesor en los días reservados para la revisión del examen teórico de junio), teniendo en cuenta que en ese caso la nota será la que obtenga en la última convocatoria (aunque dicha nota sea menor que la obtenida anteriormente).

11. Organización docente semanal (en horas presenciales del alumno)**11.1. Primer cuatrimestre:**

Semana	Horas de clases de teoría	Horas de clases de problemas	Horas de clases prácticas	Actividades Académicas Dirigidas		Horas de exámenes	Temas del temario a tratar
				Actividad	Horas		
1ª	2,0	0,0	2,0		0,0	0,0	Presentación + Tema 1
2ª	2,0	0,0	2,0		0,0	0,0	Tema 1
3ª	1,0	1,0	2,0		0,0	0,0	Problemas T1 + Tema 2
4ª	1,0	1,0	2,0		0,0	0,0	Tema 2 + Problemas T2
5ª	0,0	2,0	2,0		0,0	0,0	Problemas T2
6ª	2,0	0,0	2,0		0,0	0,0	Tema3
7ª	0,0	2,0	2,0		0,0	0,0	Problemas T3
8ª	0,0	2,0	2,0		0,0	0,0	Problemas T3
9ª	2,0	0,0	0,0	Prueba de Progreso (Temas 1 y 2)	2,0	0,0	Tema 4
10ª	1,0	1,0	2,0		0,0	0,0	Tema 4 + Problemas T4
11ª	1,0	1,0	2,0		0,0	0,0	Problemas T4 + Tema 5
12ª	2,0	0,0	2,0		0,0	0,0	Tema 5
13ª	2,0	0,0	0,0	Prueba de Progreso (Temas 3 y 4)	2,0	0,0	Tema 5
14ª	1,0	1,0	2,0		0,0	0,0	Tema 5 + Problemas T5
15ª	0,0	2,0	2,0		0,0	0,0	Problemas T5
Periodo de exámenes						6,0	
Totales	17,0	13,0	26,0		4,0	6,0	

11.2. Segundo cuatrimestre:

Semana	Horas de clases de teoría	Horas de clases de problemas	Horas de clases prácticas	Actividades Académicas Dirigidas		Horas de exámenes	Temas del temario a tratar
				Actividad	Horas		
1ª	0,0	0,0	0,0		0,0	0,0	
2ª	0,0	0,0	0,0		0,0	0,0	
3ª	0,0	0,0	0,0		0,0	0,0	
4ª	0,0	0,0	0,0		0,0	0,0	
5ª	0,0	0,0	0,0		0,0	0,0	
6ª	0,0	0,0	0,0		0,0	0,0	
7ª	0,0	0,0	0,0		0,0	0,0	
8ª	0,0	0,0	0,0		0,0	0,0	
9ª	0,0	0,0	0,0		0,0	0,0	
10ª	0,0	0,0	0,0		0,0	0,0	
11ª	0,0	0,0	0,0		0,0	0,0	
12ª	0,0	0,0	0,0		0,0	0,0	
13ª	0,0	0,0	0,0		0,0	0,0	
14ª	0,0	0,0	0,0		0,0	0,0	
15ª	0,0	0,0	0,0		0,0	0,0	
Periodo de exámenes						0,0	

Totales	0,0	0,0	0,0		0,0	0,0	
---------	-----	-----	-----	--	-----	-----	--

12. Mecanismos de control y seguimiento:

- Se propone la realización de encuestas de satisfacción entre los alumnos.
- Índices de rendimiento (tasa de éxito/fracaso, % de presentados a examen, etc.)
- Control de Asistencia