



Universidad
de Huelva

ESCUELA POLITECNICA SUPERIOR

DATOS BÁSICOS DE LA ASIGNATURA					
NOMBRE:	INFORMÁTICA				
CÓDIGO:	5027	TIPO	Optativa		
AÑO DE PLAN DE ESTUDIOS:	1999				
CRÉDITOS:	Totales		Teóricos		Prácticos
L.R.U.	6.00		3		3
E.C.T.S.	4.80				
CURSO:	Tercero	CUATRIMESTRE:	Primer Cuatrimestre	CICLO:	Primero

DATOS BÁSICOS DE LOS PROFESORES			
NOMBRE:	Francisco Alfredo Márquez Hernández <i>(Profesor para el curso 2006-2007)</i>		
CENTRO/DEPARTAMENTO:	E P S de la Rábida / Dpto. Ingeniería Electrónica, de Sistemas Informáticos y Automática		
ÁREA:	Lenguajes y Sistemas Informáticos		
Nº DE DESPACHO:	T - 40	TELÉFONO:	959217641
E-MAIL:	alfredo.marquez@diesia.uhu.es		
URL WEB:			

NOMBRE:	Nieves Pavón Pulido (Coordinadora)		
CENTRO/DEPARTAMENTO:	E P S de la Rábida / Dpto. Ingeniería Electrónica, de Sistemas Informáticos y Automática		
ÁREA:	Lenguajes y Sistemas Informáticos		
Nº DE DESPACHO:	68	TELÉFONO:	959217383
E-MAIL:	npavon@uhu.es		
URL WEB:	www.uhu.es/nieves.pavon		

DATOS ESPECÍFICOS DE LA ASIGNATURA
1. Descriptor según BOE
Ficheros. Registros. Programación de periféricos. Estructuras dinámicas de datos. Pilas. Colas. Árboles. Recursividad.

2. Situación
2.1. Prerrequisitos
Haber aprobado Fundamentos de Informática
2.2. Contexto dentro de la titulación
Asignatura optativa de 3º curso, considerada la continuación natural de Fundamentos de Informática de 1º. Interesante para ampliar los conocimientos básicos de programación estructurada necesarios para todo Ingeniero Técnico Industrial, de cualquiera de las especialidades.

2.3. Recomendaciones

Los alumnos deben asistir a las clases teóricas y de laboratorio con el fin de completar su formación en programación estructurada. Deben, a su vez, desarrollar las destrezas adquiridas de forma presencial, en un entorno no presencial. Para ello, se recomienda que los alumnos/as aprendan a obtener información útil de Internet y de la bibliografía aconsejada, sirviéndole ésta como complemento de lo aprendido en clase.

La realización de ejercicios tutorizados por el profesor, así como el desarrollo de pequeñas aplicaciones que fomenten su creatividad, serán elementos claves para la integración adecuada de esos conocimientos en su currículum.

3. Competencias que se desarrollan

3.1. Genéricas o transversales

Instrumentales:

Capacidad de análisis y síntesis.
Conocimientos de programación.
Resolución de problemas.

Personales:

Creatividad
Aprendizaje autónomo

Sistémicas

Capacidad para desarrollar aplicaciones basadas en problemas prácticos reales.

3.2. Específicas

Cognitivas(saber):

Conocimientos de informática.
Conocimientos de programación.

Procedimentales/Instrumentales(saber hacer):

Capacidad para diseñar soluciones adecuadas.
Capacidad de organización.
Capacidad de abstracción.

Actitudinales(ser):

Creatividad
Responsabilidad

4. Objetivos

Introducción de conceptos avanzados de programación: estructuras de datos complejas, programación concurrente, sistemas en tiempo real.
Introducción de conceptos avanzados acerca de la programación de sistemas para el control de dispositivos externos.
Orientación de las clases prácticas en función de la especialidad.
Hacer la asignatura lo más interesante y divertida posible.

5. Metodología

5.1. Trabajo con presencia del profesor

		Nº de horas
Clases teóricas		21
Clases prácticas		30
Exposiciones y seminarios		
Tutorías especializadas	Colectivas	
	Individuales	
Realización de actividades académicas dirigidas:		
Trabajos teóricos tutorizados		4

Diseños de pequeñas aplicaciones	5
Nº total de horas	60
5.2. Trabajo autónomo del alumno	Nº de horas
Estudio de las clases teóricas	31.5
Estudio de la clases prácticas	30
Preparación de las actividades académicas dirigidas	18.5
5.3. Realización de exámenes	Nº de horas
Realización de exámenes escritos	4
Realización de exámenes orales	0
Nº total de horas	4
Trabajo total del estudiante	144

6. Técnicas docentes
Sesiones Académicas Teóricas Sesiones Académicas Prácticas Diseño tutorizado de pequeñas aplicaciones relacionadas con la especialidad Tutorías especializadas
6.1. Desarrollo y justificación
Sesiones Académicas Teóricas Consisten en el seguimiento de clases teóricas por parte del alumno/a de forma presencial. En cada clase se desarrollan los temas expuestos en el programa de la asignatura, realizando ejemplos y ejercicios al final de cada tema con la colaboración del alumnado.
Sesiones Académicas Prácticas Consisten en la aplicación real de los conocimientos adquiridos en clase en un laboratorio destinado al efecto. Los alumnos/as aprenderán a utilizar las herramientas de programación descritas en el programa de la asignatura y realizarán aplicaciones que resuelvan problemas reales.
Diseño tutorizado de pequeñas aplicaciones relacionadas con la especialidad Consiste en la realización de aplicaciones siguiendo los esquemas diseñados por el profesor para resolver un problema de mayor calado que el de las sesiones académicas prácticas. El alumnado dispondrá de la ayuda del profesor de forma personal y on line a través de internet.
Tutorías especializadas Atención personalizada del alumno/a tanto presenciales como on line a través de internet (uso de la videoconferencia).

7. Bloques temáticos
TEMA 1. Repaso de conceptos e informática básica y programación estructurada. TEMA 2. Tipos abstractos de datos complejos: Pilas, colas y árboles binarios. TEMA 3. Gestión y control de dispositivos externos. TEMA 4. Recursividad.

8. Bibliografía
8.1. General
<ul style="list-style-type: none"> Apuntes de la asignatura en Internet http://www.uhu.es/nieves.pavon/informatica/

8.2. Específica

C++. SIN ERRORES. Chris H. Pappas; William H. Murray (Editorial McGraw-Hill). ISBN: 8448131819.

RESOLUCIÓN DE PROBLEMAS CON C++. Walter Savitch : University of California at San Diego (Addison Wesley) . ISBN: 9684444168.

C++ PARA INGENIERÍA Y CIENCIAS. G. J. BRONSON (Paraninfo) ISBN: 9687529873. (1999).

NUMERICAL RECIPES IN C++. Press, William H.; Teukolsky, Saul A.; Vetterling, William T.; Flannery, Brian P. (Cambridge University Pr.) ISBN: 0521750334. 2ª edición (2002).

9. Técnicas de evaluación

Examen Teórico Escrito. Consistente en una prueba donde el alumno deberá resolver distintos problemas y/o cuestiones teóricas referentes a los temas teóricos desarrollados durante el cuatrimestre.

Se realizará una evaluación continua de la asignatura.

Cada tema se explicará y se hará un control de conocimientos que servirá para la nota final.

Sesiones de problemas. En todas las actividades formativas el profesor valorará la participación activa del alumno. Se valorará de forma directa tanto en las sesiones de problemas como en las del laboratorio.

Se desarrollará una pequeña aplicación como práctica final.

9.1. Criterios de evaluación y clasificación

Al final de cada tema se hará un control de conocimientos.

Si el alumno/a obtiene una nota media entre los 4 controles ≥ 5 no tendrá que presentarse al examen final.

La nota media de los 4 controles o en su defecto la nota del examen final debe ser ≥ 5 para aprobar la parte teórica de la asignatura.

La nota de la parte práctica debe ser ≥ 5 para aprobar dicha parte.

Sólo se hace media entre ambas partes si están aprobadas por separado.

La nota de teoría pesa 7,5 puntos y la nota de prácticas 2,5.

Las notas de teoría y prácticas se guardan por separado hasta septiembre.

10. Organización docente semanal

(Sólo indicar el número de horas que a ese tipo de sesión va a dedicar el estudiante cada semana)

Nº de horas 1 ^{er} Cuatr.	Sesiones Teóricas	Sesiones Prácticas (problemas)	Sesiones Prácticas	Exposiciones y Seminarios	Tutorías Especializadas Colectivas	Otras actividades		Exámenes	Temas de teoría a tratar
						Horas	Actividad		
1 ^a Semana:3-7 de Oct .2005	2		2						Tema 1
2 ^a :10-12-14 de Octubre	2		2						Tema 1
3 ^a :17-21 Octubre	1	1	2						Tema 2 Punto 1
4 ^a : 24-28 de Octubre	1	1	2						Tema 2 Punto 2
5 ^a : 31Oct-1Nov-4Nov	1	1	2						Tema 2 Punto 3
6 ^a : 7-11 de Noviembre			2			2	Trabajo teórico		
7 ^a : 14-18 de Nov.			2			2	Trabajo teórico		
8 ^a : 21-25 de Nov	2		2						Tema 3 Introducción
9 ^a : 28Nov-2Dic.	1		2			1	Diseño aplicación		Tema 3 Punto 1
10 ^a : 5-6-8-9Dic									
11 ^a : 12-16 de Dic.			2			2	Diseño aplicación		
12 ^a :19-22 de Dic.			2			2	Diseño aplicación		
13 ^a : 26-30 de Dic.									
14 ^a :2-6 de Ene. 2006									
15 ^a : 9-13 de Enero	2		2						Tema 3 Puntos 2 - 3
16 ^a : 16-20 de Enero	2		2						Tema 4 Puntos 1-2-2
17 ^a : 23-27 de Enero	1	1	2						Tema 4 Resto puntos
18 ^a :30Ene-	0	2	2						Repaso

3Feb.									
19 ^º : 6-10 de Febrero								2	Examen teoría
20 ^º :13-17 de Feb.								2	Examen práctica

Distribuya el número de horas que ha respondido en el punto 5 en 15 semanas para una asignatura cuatrimestral y 30 anuales
Las semanas en rojo son no lectivas y las grises son de exámenes.

11. Temario desarrollado

TEMA 1. Repaso de conceptos e informática básica y programación estructurada.

- 1.1. Introducción.
- 1.2. Aspectos relacionados con el hardware.
- 1.3. Aspectos relacionados con el software.
 - 1.3.1. El sistema operativo.
 - 1.3.1.1. Sistemas basados en MS-DOS.
 - 1.3.1.2. Sistemas basados en Windows.
 - 1.3.1.3. Sistemas basados en Linux.
 - 1.3.2. Programación de aplicaciones.
 - 1.3.2.1. Concepto de Ingeniería del Software.
 - 1.3.2.2. Programación estructurada. Diseño descendente.
 - 1.3.2.3. Lenguajes de programación. Niveles de abstracción.
 - 1.3.2.4. Repaso del lenguaje C.
- 1.4. Ejercicios.
- 1.5. Extensiones del lenguaje C: Lenguaje C++.

TEMA 2. Tipos abstractos de datos complejos: Pilas, colas y árboles binarios.

- 2.1. Tipo abstracto de datos PILA.
 - 2.1.1. Definición y operaciones.
 - 2.1.2. Implementaciones.
- 2.2. Tipo abstracto de datos COLA.
 - 2.2.1. Definición y operaciones.
 - 2.2.2. Implementaciones.
- 2.3. Tipo abstracto de datos ÁRBOL BINARIO.
 - 2.3.1. Definición y operaciones.
 - 2.3.2. Implementaciones.

TEMA 3. Gestión y control de dispositivos externos.

- 3.1. Acceso a dispositivos hardware mediante puertos en MS-DOS.
- 3.2. Gestión de dispositivos en sistemas basados en Windows.
- 3.3. Gestión de dispositivos en sistemas basados en Linux.

TEMA 4. Recursividad.

- 4.1. Introducción a la recursividad. Definición de algoritmo recursivo.
- 4.2. Recursividad directa.
- 4.3. Recursividad indirecta.
- 4.4. Ejemplos de problemas recursivos.
- 4.5. Problemas derivados del uso de la recursividad.
- 4.6. Soluciones mediante backtracking.
- 4.7. Ejercicios.

12. Mecanismos de control y seguimiento

Al final de cada tema se hará un control de conocimientos.

Si el alumno/a obtiene una nota media entre los 4 controles ≥ 5 no tendrá que presentarse al examen final.