

Grado en Ingeniería Informática

DATOS DE LA ASIGNATURA

Nombre:				
Estructuras de Datos I				
Denominación en inglés:				
Data structures I				
Código:		Carácter:		
606010109		Básico		
Horas:				
	Totales	Presenciales	No presenciales	
Trabajo estimado:	150	60	90	
Créditos:				
	Grupos reducidos			
Grupos grandes	Aula estándar	Laboratorio	Prácticas de campo	Aula de informática
3	0	0	0	3
Departamentos:		Áreas de Conocimiento:		
Tecnologías de la Información		Lenguaje y Sistemas Informáticos		
Curso:		Cuatrimestre:		
1º - Primero		Segundo cuatrimestre		

DATOS DE LOS PROFESORES

Nombre:	E-Mail:	Teléfono:	Despacho:
*Ortíz Sañudo, Lourdes	lourdes@uhu.es	959217391	TUPB - 44
Fernández Bejarano, Francisco Javier	javier.fernandez@dti.uhu.es	959217648	TUPB-28

*Profesor coordinador de la asignatura

1. Descripción de contenidos

1.1. Breve descripción (en castellano):

- Tipos abstractos de datos (TAD)
- TAD genéricos
- Especificación, implementación y manipulación de Tipos Abstractos de Datos lineales: listas, pilas y colas
- Manejo de archivos

1.2. Breve descripción (en inglés):

- Abstract data types (ADT)
- Generic ADTs
- Specification, implementation and handling of linear ADTs: lists, stacks and queues
- File handling

2. Situación de la asignatura

2.1. Contexto dentro de la titulación:

- La asignatura es impartida en el segundo cuatrimestre de primer curso del grado de Ingeniería Informática. Su duración aproximada es de 15 semanas de docencia comprendida desde los meses de Febrero a Junio.
- La asignatura es continuada y sirve como base de la asignatura *Estructuras de Datos II* que es impartida en el segundo cuatrimestre de segundo curso del grado de Ingeniería Informática.
- La asignatura se basa en los conocimientos adquiridos en la asignatura *Fundamentos de Programación*, que es impartida en el primer cuatrimestre de primero de Ingeniería Informática.

2.2. Recomendaciones:

- Aunque esta asignatura es base para otras asignaturas de la titulación y aun siendo de primer curso, es muy recomendable que el alumno tenga sólidamente adquiridos los conocimientos impartidos en la asignatura *Fundamentos de Programación*.
- Es muy recomendable que el alumno conozca y sepa:
 - Confeccionar en un lenguaje estructurado y dirigido a objetos, algoritmos correctos que resuelvan un problema de pequeña-mediana envergadura, expuesto en términos de una especificación más o menos formal, y decidir cuál de las posibles soluciones es la más apropiada para un entorno determinado.
 - Codificar de una manera correcta mediante el uso de estructuras de control claras, bucles, sentencias condicionales, etc., según convenga a la claridad y finalidad del segmento de código.
 - Agrupar conjuntos de instrucciones algorítmicas coherentemente mediante funciones y/o métodos.
 - Los tipos de datos básicos que ofrece cualquier lenguaje de programación.
 - El mecanismo básico de paso de parámetros y utilizarlo correctamente.
 - Aplicar algunas técnicas algorítmicas sencillas para la resolución de problemas.

3. Objetivos (Expresados como resultados del aprendizaje):

Podemos considerar los siguientes objetivos de carácter general:

- Conocer como organizar un determinado volumen de datos de la forma más racional posible en función de los requisitos del problema a resolver.
- Conocer los mecanismos de abstracción y su importancia para la resolución de problemas tanto recursivos como no recursivos..
- Conocer el funcionamiento y las técnicas básicas de ordenación de la información y la consulta eficiente de la misma.
- Conocer el funcionamiento y la utilidad de la gestión de memoria masiva auxiliar.
- Comprender la necesidad de separación entre los niveles de especificación, implementación y su aplicación en el desarrollo software.
- Saber analizar, especificar y documentar tipos abstractos de datos.
- Conocer los conceptos de programación basada en tipos abstractos.
- Conocer los tipos abstractos de datos más usuales en programación, sus implementaciones más comunes y su utilidad.
- Desarrollar programas, basándose en tipos abstractos de datos mediante el lenguaje orientado a objetos C++.
- Ser capaz de implementar de diferentes formas una especificación de software dada. El alumno debe saber escoger entre diferentes implementaciones alternativas de una abstracción de datos, y razonar sobre la solución escogida en función de los recursos necesarios (tiempo de ejecución, espacio requerido, etc.).
- Saber resolver problemas utilizando los TAD más apropiados.
- Entender el funcionamiento y la utilidad de la gestión dinámica de memoria.
- Diseñar e implementar software capaz de manejar volúmenes dinámicos de información de manera eficiente, determinando en cada caso la estructura dinámica de datos más eficiente y adecuada.

4. Competencias a adquirir por los estudiantes

4.1. Competencias específicas:

- **CB03:** Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para la resolución de problemas propios de la ingeniería
- **CB04:** Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería
- **CB05:** Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

4.2. Competencias básicas, generales o transversales:

- **CB1:** 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
- **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
- **CG0:** Capacidad de análisis y síntesis: Encontrar, analizar, criticar (razonamiento crítico), relacionar, estructurar y sintetizar información proveniente de diversas fuentes, así como integrar ideas y conocimientos.
- **G02:** Capacidad de comunicación oral y escrita en el ámbito académico y profesional con especial énfasis, en la redacción de documentación técnica
- **G03:** Capacidad para la resolución de problemas
- **G05:** Capacidad de trabajo en equipo.
- **T02:** Conocimiento y perfeccionamiento en el ámbito de las TIC's

5. Actividades Formativas y Metodologías Docentes

5.1. Actividades formativas:

- Sesiones de Teoría sobre los contenidos del Programa.
- Sesiones de Resolución de Problemas.
- Sesiones Prácticas en Laboratorios Especializados o en Aulas de Informática.

5.2. Metodologías docentes:

- Clase Magistral Participativa.
- Desarrollo de Prácticas en Laboratorios Especializados o Aulas de Informática en grupos reducidos.
- Resolución de Problemas y Ejercicios Prácticos.
- Evaluaciones y Exámenes.

5.3. Desarrollo y justificación:

Sesiones teóricas

- La metodología de enseñanza que se va a seguir a lo largo de la asignatura será mediante clases magistrales (teórico-prácticas) en el aula asignada, compaginadas con las clases prácticas de laboratorio que consistirán en la implementación, por parte del alumno, de software correspondiente a los contenidos impartidos en la teoría.
- En las clases teóricas se presentarán los conceptos de manera clara y concisa utilizando para ello las herramientas docentes más adecuadas al alcance del profesor.

Sesiones de resolución de problemas

- Problemas en grupos reducidos, en el que se desarrollarán ejercicios, se resolverán dudas y en las que se podrán evaluar los conocimientos de cada alumno mediante preguntas y/o cuestionarios.

Sesiones prácticas

- Las prácticas de laboratorio serán realizadas en las aulas de informática. Todas las prácticas tendrán una duración directamente proporcional a la complejidad de las mismas.

6. Temario desarrollado:

Tema 1. Punteros y manejo de archivos

1. Punteros
2. Archivos lógicos y físicos
3. Tipos de archivos
4. Operaciones con archivos
5. Consideraciones sobre el acceso secuencial y directo

Tema 2. Tipos abstractos de datos

1. Abstracción de datos y Tipos Abstractos de Datos (TAD)
2. Especificación algebraica de un TAD
3. Construcción de especificaciones

Tema 3. Tipos abstractos de datos lineales

1. TAD Lista
2. TAD Pila
3. TAD Cola

Tema 4. TAD Lineales mediante memoria dinámica

1. Asignación dinámica de memoria
2. TAD lineales con memoria dinámica
3. Paso de parámetros por puntero

Tema 5. TAD Genéricos

1. Introducción
2. Plantillas de funciones o funciones genéricas
3. Tipos genéricos

7. Bibliografía

7.1. Bibliografía básica:

- Fernández F.J., Martín J.M., Ortiz L., Roche F., Roldán A., *Estructuras de Datos en C++*. Ed. Servicio de Publicaciones de la Universidad de Huelva
- Luis Joyanes Aguilar. *Fundamentos de Programación. Algoritmos, Estructuras de Datos y Objetos*. Ed : McGraw-Hill
- N. Wirth. *Algoritmos + Estructuras de Datos = Programas*. Ed. Ediciones del Castillo

7.2. Bibliografía complementaria:

- B. Stroustrup. *El Lenguaje de Programación C++*. Ed. Addison Wesley

8. Sistemas y criterios de evaluación.

8.1. Sistemas de evaluación:

- Examen de teoría/problemas
- Examen de prácticas

8.2. Criterios de evaluación y calificación:

Teniendo en cuenta que esta asignatura es eminentemente práctica, todas las competencias CB03, CB04, CB05 CB1, CB3, CG0, G02, G03 y T02 se adquirirán con los sistemas de evaluación especificados en esta guía, a excepción de la competencia G05 (Capacidad de trabajo en equipo) que se evaluará en la parte práctica de la asignatura.

Evaluación de la parte teórica (TEORÍA)

- Convocatoria de Junio: La parte teórica de la asignatura tendrá un valor de 5 puntos (50% de la nota final) evaluados mediante un examen realizado en la convocatoria oficial de junio. La nota obtenida se guarda para la convocatoria de septiembre si: (Nota Teoría + Nota Problemas) evaluada sobre 7 puntos es mayor o igual a 3.5.
- Convocatoria de Septiembre: La parte teórica de la asignatura tendrá un valor de 7 puntos (70% de la nota final) evaluados mediante un examen realizado en la convocatoria oficial de septiembre.

Evaluación de las sesiones de resolución de problemas (PROBLEMAS)

- Convocatoria de Junio: Las pruebas de resolución de problemas que el alumno debe realizar tendrán un valor máximo de 2 puntos (20% de la nota final de la asignatura). La nota de estas pruebas no se guardará para convocatorias futuras.
- Convocatoria de Septiembre: Estas pruebas no se evalúan por ser actividades realizadas durante el periodo de docencia en el cuatrimestre.

Evaluación de la parte práctica (PRÁCTICAS)

- Convocatoria de Junio: Las prácticas de la asignatura tendrán un valor de 3 puntos (30% de la nota final) repartidos en una o varias pruebas durante el cuatrimestre. La nota obtenida se guarda para la convocatoria de septiembre.
- Convocatoria de Septiembre: Las prácticas de la asignatura tendrán un valor de 3 puntos (30% de la nota final) evaluados en una prueba. El alumno podrá presentarse a la convocatoria de septiembre independientemente de la puntuación obtenida en la convocatoria de junio, pero teniendo en cuenta que siempre se utilizará para la nota final el valor del último examen realizado.

Nota final en Junio:

Si (Nota TEORÍA + Nota PROBLEMAS) \geq 3.5 :

Nota FINAL = Nota TEORÍA + Nota PROBLEMAS + Nota PRÁCTICAS

En caso contrario:

Nota FINAL = Nota TEORÍA + Nota PROBLEMAS

(Nota TEORÍA se evalúa sobre 5 puntos)

Nota final en Septiembre:

Si (Nota TEORÍA) \geq 3.5 :

Nota FINAL = Nota TEORÍA + Nota PRÁCTICAS

En caso contrario:

Nota FINAL = Nota TEORÍA

(Nota TEORÍA se evalúa sobre 7 puntos)

9. Organización docente semanal orientativa:

	Semanas	Grupos Grandes	Grupos Reducidos Aula Estándar	Grupos Reducidos Aula de Informática	Grupos Reducidos Laboratorio	Grupos Reducidos prácticas de campo	Pruebas y/o actividades evaluables	Contenido desarrollado
#1	2	0	2	0	0		Tema 1	
#2	2	0	2	0	0		Tema 1	
#3	2	0	2	0	0		Tema 2	
#4	2	0	2	0	0		Temas 2 y 3	
#5	2	0	2	0	0		Tema 3	
#6	2	0	2	0	0		Tema 3	
#7	2	0	2	0	0		Tema 3	
#8	2	0	2	0	0		Tema 3	
#9	2	0	2	0	0	Primera prueba práctica	Tema 4	
#10	2	0	2	0	0		Tema 4	
#11	2	0	2	0	0		Tema 4	
#12	2	0	2	0	0		Tema 4	
#13	2	0	2	0	0		Tema 4	
#14	2	0	2	0	0		Temas 4 y 5	
#15	2	0	2	0	0	Segunda prueba práctica	Tema 5	
	30	0	30	0	0			