

## Guía docente

# 320099 - EDOO - Estructuras de Datos y Orientación a Objetos

Última modificación: 23/04/2019

**Unidad responsable:** Escuela Superior de Ingenierías Industrial, Aeroespacial y Audiovisual de Terrassa

**Unidad que imparte:** 723 - CS - Departamento de Ciencias de la Computación.

**Titulación:** GRADO EN INGENIERÍA DE SISTEMAS AUDIOVISUALES (Plan 2009). (Asignatura obligatoria).

**Curso:** 2019      **Créditos ECTS:** 6.0      **Idiomas:** Catalán

### PROFESORADO

---

**Profesorado responsable:** Pau Fernández

**Otros:** Pau FernándezJordi MarcoPepa López

### CAPACIDADES PREVIAS

---

Se considera muy conveniente haber aprobado el curso sobre Fundamentos de Informática del primer cuatrimestre.

### COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

---

**Específicas:**

1. Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.

**Transversales:**

2. TRABAJO EN EQUIPO - Nivel 1: Participar en el trabajo en equipo y colaborar, una vez identificados los objetivos y las responsabilidades colectivas e individuales, y decidir conjuntamente la estrategia que se debe seguir.

### METODOLOGÍAS DOCENTES

---

Tipo de sesiones:- Sesiones presenciales de exposición de los contenidos (utilizando unos apuntes estándar, repartidos previamente).- Sesiones presenciales de trabajo práctico al laboratorio (siguiendo un guión con ejercicios intercalados).- Trabajo autónomo de estudio y realización de ejercicios.- Preparación y realización de actividades evaluables en grupo.En las sesiones de exposición de los contenidos el profesor introducirá las bases teóricas de la materia, conceptos, métodos y resultados ilustrándolos con ejemplos convenientes por facilitar su comprensión.Las sesiones de trabajo práctico al aula serán de tres clases:a) Sesiones en las que los estudiantes seguirán un guión de prácticas con ejercicios intercalados y el profesor resolverá las dudas que vayan surgiendo.b) Sesiones de presentación de trabajos realizados en grupo por parte de los estudiantes.c) Sesiones de exámenes.Los estudiantes, de forma autónoma deberán estudiar por tal de asimilar los conceptos, resolver los ejercicios propuestos ya sea con lápiz y papel o usando el ordenador.Los estudiantes elaborarán un proyecto de programación en grupos de dos o tres.

### OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

---

La asignatura tiene dos objetivos fundamentales:- Dar al estudiante herramientas para controlar la complejidad en la construcción de programas (vía la orientación a objetos).- Preparar al estudiante para diseñar estructuras de datos eficientes en un amplio abanico de problemas de programación.



## HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTADO

Tipo	Horas	Porcentaje
Horas grupo grande	30	20.00
Horas aprendizaje autónomo	90	60.00
Horas grupo pequeño	30	20.00

**Dedicación total:** 150 h

## CONTENIDOS

### Eficiencia de Algoritmos

**Descripción:**

2.1. Órdenes de crecimiento 2.2. Notación asintótica 2.3. Análisis de algoritmos

**Objetivos específicos:**

- Enumerar los órdenes de crecimiento típicos.- Estimar el orden de crecimiento de programas sencillos.

**Actividades vinculadas:**

2 actividades de Tipo 1

### Estructuras de Datos

**Descripción:**

3.1. Estructuras secuenciales: vector, lista, pila y cola. 3.2. Estructuras de tipo conjunto. 3.3. Estructuras asociativas (tablas asociativas). 3.4. Implementación de estructuras. 3.5. Eficiencia de operaciones típicas. 3.6. Comparación entre estructuras. 3.7. Algoritmos sobre estructuras de datos. 3.8. Librerías de estructuras a datos (STL).

**Objetivos específicos:**

- Declarar estructuras de cualquier tipo.- Acceder a los elementos de una estructura eficientemente.- Insertar, borrar y modificar los elementos de cualquier estructura eficientemente.- Buscar y recorrer los elementos en cualquier estructura eficientemente.- Utilizar iteradores de forma eficiente para cada estructura.- Escoger las estructuras más eficientes para un problema dado.- Desarrollar programas utilizando librerías de estructuras.

**Actividades vinculadas:**

3 actividades de Tipo 1 1 actividad de Tipo 2 Finalización de la actividad de Tipo 3

### Clases y Objetos

**Descripción:**

1.1. Barreras de abstracción 1.2. Tipos abstractos de datos (TADs) 1.3. Especificación e implementación 1.4. Miembros: atributos y métodos 1.5. Accesibilidad a miembros 1.6. Tipos de métodos: constructores, destructores, consultores, modificadores y operadores 1.7. Sobrecarga de métodos 1.8. Gestión dinámica de la memoria

**Objetivos específicos:**

- Diseñar un conjunto de operaciones para un TAD.- Determinar los constructores necesarios para una clase.- Distinguir los diferentes tipos de métodos.- Implementar constructores de clases simples y agregadas.- Detectar errores en acceso a miembros.- Distinguir el método escogido en una llamada sobrecargada.- Crear objetos en memoria dinámica.

**Actividades vinculadas:**

4 actividades de Tipo 1 1 actividad de Tipo 2

## ACTIVIDADES

### TIPO 3: TAREA INDIVIDUAL DE EVALUACIÓN CONTINUA A

**Descripción:**

Realización de un ejercicio individual sobre el contenido del tema, fuera del aula. Corrección por parte del profesorado. Posteriormente, el profesorado durante la sesión siguiente llevará a cabo una reflexión general en el aula sobre los errores más comunes de este ejercicio.

**Objetivos específicos:**

Los correspondientes al tema que se está trabajando con el ejercicio.

**Material:**

Apuntes del tema disponibles en Atenea.

**Entregable:**

Resolución del ejercicio entregable a Atenea. Las actividades de tipo 3 puntúan tienen un peso de un 10% en conjunto.

**Dedicación:** 3 h

Aprendizaje autónomo: 3h

### TIPO 1: CONTROL DE LABORATORIO

**Descripción:**

Realización de un ejercicio individual en el aula que cubra todos los objetivos específicos. Corrección por parte del profesorado.

**Material:**

Apuntes del tema elaborados por el profesorado (Atenea). Enunciado del ejercicio. Resolución del ejercicio. Criterios de corrección (rúbrica).

**Entregable:**

Resolución del ejercicio por parte del estudiante / a. Todas las actividades de tipo 1 corresponden al 20% de la asignatura (Laboratorio).

**Dedicación:** 2 h

Grupo pequeño/Laboratorio: 2h

### PROYECTO EN GRUPO DE EVALUACIÓN CONTINUA

**Descripción:**

Realización de un proyecto de programación que incluya todos los objetivos específicos del curso. Se realiza en grupo. Corrección por parte del profesorado.

**Material:**

Apuntes de la asignatura (que incluyan ejercicios resueltos de una dificultad similar).

**Entregable:**

Entregables: uno para cada fase del proyecto (habrá varias, especificadas en el enunciado). La nota de esta actividad corresponde al 20% de la asignatura (Proyecto).

**Dedicación:** 45 h

Grupo mediano/Prácticas: 39h

Actividades dirigidas: 6h



## SISTEMA DE CALIFICACIÓN

---

Examen parcial (P): 20%Examen final (F): 30%Controles (C1, C2): 20%Resolución de problemas (T): 10%Proyecto (J): 20% Tanto los controles como el examen parcial no podrán bajar notas posteriores. La fórmula para calcular la nota final es, de hecho:  $EDOO = 0.2 * J + 0.1 * T + \text{MAX}(0.7 * F, \text{MAX}(0.3 * P, 0.2 * P + 0.1 * C1) + \text{MAX}(0.4 * F, 0.3 * F + 0.1 * C2))$  Para aquellos estudiantes que cumplan los requisitos y se presenten al examen de reevaluación, la calificación del examen de reevaluación substituirá las notas de todos los actos de evaluación que sean pruebas escritas presenciales (controles, exámenes parciales y finales) y se mantendrán las calificaciones de prácticas, trabajos, proyectos y presentaciones obtenidas durante el curso. Si la nota final después de la reevaluación es inferior a 5.0 substituirá la inicial únicamente en el caso de que sea superior. Si la nota final después de la reevaluación es superior o igual a 5.0, la nota final de la asignatura será aprobado 5.0.

## BIBLIOGRAFÍA

---

### Básica:

- Stroustrup, B. The C++ programming language. 3rd ed. Reading: Addison-Wesley, 1997. ISBN 0201889544.
- Josuttis, Nicolai M. The C++ standard library. 2nd ed. Massachusetts: Addison-Wesley, 2012. ISBN 0201379260.

### Complementaria:

- Meyers, Scott. Effective STL : 50 specific ways to improve your use of the Standard Template Library. Boston: Addison-Wesley, 2001. ISBN 0201749629.