
Curso Académico: (2020 / 2021)**Fecha de revisión: 10/07/2020 12:32:34**

Departamento asignado a la asignatura:**Coordinador/a: VALLS FERRAN, JOSE MARIA****Tipo: Optativa Créditos ECTS : 6.0****Curso : 4 Cuatrimestre :**

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

Programación, Estructura de Datos y Algoritmos

OBJETIVOS

Conocimiento y aplicación de los procedimientos algorítmicos básicos de las tecnologías informáticas para diseñar soluciones a problemas, analizando la idoneidad y complejidad de los algoritmos propuestos. (CECRI6)

Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados. (CECRI8)

DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

Tema 1. Introducción a la Programación Orientada a Objetos. Modelado

- 1.1. Introducción a la POO
- 1.2. Comparación con otros paradigmas de programación.
- 1.3. Modelado Orientado a Objetos.

Tema 2. Clases y Objetos en Java. Herencia y polimorfismo

- 2.1 Composición y herencia
- 2.2 Clases abstractas
- 2.3 Herencia múltiple. Interfaces
- 2.4 Polimorfismo

Tema 3. Excepciones, clases de utilidad y paquetes. Entrada y Salida

- 3.1 Excepciones
- 3.2 Clases de utilidad
- 3.3 Documentación y Paquetes
- 3.4 E/S estándar. Ficheros en Java

Tema 4. Algoritmos avanzados

- 4.1 Complejidad Computacional
- 4.2 Algoritmos recursivos
- 4.3 Algoritmos de búsqueda
- 4.4 Algoritmos avanzados de ordenación
- 4.5 Diseño de algoritmos complejos con Orientación a Objetos

Tema 5. Patrones de diseño. Otros lenguajes orientados a objetos

- 5.1 Introducción.
- 5.2 Catálogo de patrones de diseño
- 5.3 Otros lenguajes orientados a objetos

ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

Esta asignatura optativa no se oferta en el curso 20-21 con lo que no se especifican las actividades especiales debidas a COVID-19, sino solamente las actividades previstas en una situación de normalidad.

- Clases magistrales. La asignatura se impartirá mediante lecciones magistrales para dar a conocer los conceptos fundamentales y avanzados de la POO, los algoritmos complejos y las técnicas de diseño. Se motivará la participación activa de los alumnos y se incentivará la discusión de forma que se argumente sobre los motivos en utilizar determinadas técnicas de programación para resolver determinados problemas. 3 ECTS

- Clases prácticas. Las clases de teoría se reforzarán con sesiones de tipo práctico en aulas informáticas donde se realizarán ejercicios que permitirán fijar y aplicar los conocimientos teóricos adquiridos, así como las prácticas evaluables. 3ECTS

SISTEMA DE EVALUACIÓN

Peso porcentual del Examen/Prueba Final: 60

Peso porcentual del resto de la evaluación: 40

La evaluación de la asignatura se hará mediante evaluación continua y un examen final.

La evaluación continua permitirá a los alumnos seguir el proceso de aprendizaje y obtener a lo largo del cuatrimestre el 40% de la calificación final (4 puntos). Esta evaluación se realizará a través de dos prácticas con un valor del 30% de la nota final y de un examen parcial con el 10% de la nota final.

Al examen final le corresponderá al 60% de la nota final (6 puntos) y contendrá una parte de ejercicios de tipo test y una parte de resolución de problemas de programación.

El examen final es obligatorio y se deberá obtener un mínimo de 2,5 puntos (sobre los 6 puntos del examen) para poder aprobar la asignatura.

La calificación final de la asignatura se obtendrá sumando la calificación obtenida por la evaluación continua y la obtenida en el examen final.

Si el alumno no ha realizado la evaluación continua, podrá realizar el examen final mediante el que podrá obtener el 60% de la nota final.

En la convocatoria extraordinaria, el examen tendrá el mismo peso que en la convocatoria ordinaria, manteniéndose la nota obtenida en la evaluación continua. Si el alumno no ha realizado la evaluación continua tendrá derecho a realizar un examen con un valor del 100% de la nota final. Aunque el alumno haya realizado la evaluación continua podrá renunciar a ella si esta opción le es más favorable.

BIBLIOGRAFÍA BÁSICA

- Bruce Eckel Piensa en Java, Pearson/Prentice Hall , 2007

- David Camacho, José M^a Valls, Jesús García, José M. Molina, Enrique Bueno Programación, algoritmos y ejercicios resueltos en Java, Pearson/Prentice Hall, 2003

BIBLIOGRAFÍA COMPLEMENTARIA

- Brassard, Giles Fundamentos de Algoritmia, Prentice hall international, 1997

- Gamma Erich Patrones de diseño, Addison Wesley, 2003