

Curso Académico: (2018 / 2019)

Fecha de revisión: 09/05/2018 12:18:34

Departamento asignado a la asignatura: Departamento de Informática

Coordinador/a: SEGURA BEDMAR, ISABEL

Tipo: Formación Básica Créditos ECTS : 6.0

Curso : 1 Cuatrimestre : 2

Rama de Conocimiento: Ingeniería y Arquitectura

REQUISITOS (ASIGNATURAS O MATERIAS CUYO CONOCIMIENTO SE PRESUPONE)

- Programación.
- Cálculo

OBJETIVOS

COMPETENCIA Nº CRÉDITOS ECTS EVIDENCIAS DE EVALUACIÓN
 CECRI1 (1 ECTS), Evidencia de Evaluación: PRÁCTICAS DE LABORATORIO, TRABAJOS COLECTIVOS (CASO PRÁCTICO)
 CECRI6 (ALGORITMOS) (1.5 ECTS) Evidencia de Evaluación: EXÁMENES, PRÁCTICAS DE LABORATORIO, TRABAJO COLECTIVO E INDIVIDUAL
 CECRI7 (TIPOS Y ESTRUCTURAS DE DATOS) (2.5 ECTS), Evidencia de Evaluación: EXÁMENES, PRÁCTICAS DE LABORATORIO, TRABAJO COLECTIVO E INDIVIDUAL
 CGB4 (0,50 ECTS), Evidencia de Evaluación: PRÁCTICAS DE LABORATORIO, TRABAJOS COLECTIVOS
 CGB% (0,50 ECTS) Evidencia de Evaluación: PRÁCTICAS DE LABORATORIO, TRABAJO COLECTIVO

1. Competencias Transversales/Genéricas

- Capacidad de análisis y síntesis (PO e)
- Capacidad de organizar y planificar (PO d)
- Resolución de problemas (PO e)
- Trabajo en equipo (PO d)
- Capacidad de aplicar los conocimientos en la práctica (PO e)

2. Competencias Específicas

a. Cognitivas (Saber)

- Conocimiento general sobre algorítmica (PO a)
- Conocimiento de estructuras de datos fundamentales (PO k)
- Conocimiento de estructuras de datos avanzadas (PO k)

b. Procedimentales/Instrumentales (Saber hacer)

- Diseñar y analizar la complejidad de algoritmos (PO a)
- Interpretar y utilizar las diferentes estructuras de datos (PO k)
- Implementar soluciones a problemas específicos mediante estas herramientas (PO e)

c. Actitudinales (Ser)

- Capacidad de resolución de problemas mediante algoritmos (PO e)
- Preocupación por la claridad, sencillez y eficiencia en la resolución de problemas (PO e y k)
- Debatir y concluir las distintas soluciones a un problema (PO e y k)

Los resultados de aprendizaje son:

Trabajo individual:

- Resolución por parte del alumno de problemas en el que debe demostrar que tienen capacidad para combinar la teoría y práctica (PO a, e, k)

Trabajo en grupo:

Caso Práctico sobre diseño e implementación de estructuras de datos. (PO a, d, e, k)

DESCRIPCIÓN DE CONTENIDOS: PROGRAMA

1. Introducción
 - a. Concepto Tipo Abstracto de Datos (TAD).
 - b. Especificación e Implementación de TAD
2. Tipos Abstractos de Datos lineales
 - a. Concepto TAD Lineal. Estructura estática vs dinámica.
 - b. Pilas.
 - c. Colas.
 - d. Listas: simples y dobles.
3. Algoritmos I: Análisis de la Complejidad.
 - a. Análisis de Algoritmos
 - b. Tipos de Complejidad
 - c. Función Tiempo de Ejecución.
 - d. Orden de Complejidad
 - e. Caso peor, mejor y promedio.
4. Algoritmos II: Recursión.
5. Tipos Abstractos de Datos Jerárquicos: Árboles
 - a. Árboles generales
 - b. Árboles binarios
 - c. Recorrido de un árbol: preorden, en orden, postorden, por niveles
 - d. Árboles Binarios de búsqueda.
 - e. ABB equilibrados.
6. Grafos.
 - a. Definición TAD Grafo. Aplicaciones.
 - b. Implementación basada en matriz de adyacencias.
 - c. Implementación basada en lista de adyacencias.
 - d. Recorridos en profundidad y amplitud
 - e. Algoritmo de Dijkstra (camino más corto).
7. Algoritmos III.
 - a. Divide y Venceras.
 - b. Panorámica de las principales estrategias algorítmicas.

ACTIVIDADES FORMATIVAS, METODOLOGÍA A UTILIZAR Y RÉGIMEN DE TUTORÍAS

1. Clases Teóricas: Tienen por objetivo alcanzar las competencias específicas cognitivas de la asignatura (PO a y k)
 2. Actividades académicas dirigidas por el profesor:
 - 2.1. Con presencia del profesor: Resolución de ejercicios de análisis, diseño e implementación con distinto nivel de complejidad en colaboración con los alumnos (PO a, e). Parte de estas actividades se llevarán a cabo en laboratorio informático (PO k).
 - 2.2. Sin presencia del profesor: Ejercicios, prácticas de implementación y lecturas básicas y complementarias propuestas por el profesor (PO k, e)
- Además estas actividades pueden realizarse como:
- a). Trabajo individual: Consistente en el desarrollo de las soluciones a los problemas y ejercicios propuestos por el profesor.
 - b). Trabajo en grupo: Consistente en el desarrollo de las soluciones a los problemas colectivos propuestos por el profesor (PO d).

3. Prueba de evaluación intermedia y examen final de la asignatura (PO a, e, k)
4. Se realizará una tutoría colectiva por cada grupo pequeño para resolver las preguntas y dudas de los estudiantes.

SISTEMA DE EVALUACIÓN

Peso porcentual del Examen/Prueba Final:	60
Peso porcentual del resto de la evaluación:	40

Los ejercicios y exámenes además de servir como actividad formativa tienen el doble objetivo de ser medida para el sistema de evaluación. Durante el curso, se publicarán dos hojas de ejercicios en aula global. Los alumnos también deberán tratar de resolver un caso práctico aplicando los conceptos explicados durante el curso.

Se seguirá un sistema de evaluación continua que incluye la valoración de las actividades académicas dirigidas y prácticas así como la prueba de evaluación intermedia y el examen final según la siguiente ponderación:

* Prueba de evaluación intermedia: 20% (Po a, e, k)

* Resolución Caso Práctico: 20% (Po a, e, d, k). El caso práctico incluye la realización de pruebas unitarias utilizando bibliotecas de evaluación (por ejemplo JUnit en Java o unittest en Python).

* Problemas semanales: Durante las 5-6 primeras semanas del semestre, se publicará un problema de forma semanal, que el alumno deberá intentar resolver de forma individual y entregar su solución a través de AulaGlobal antes de los 7 días naturales de su publicación. Si el alumno entrega y resuelve correctamente al menos un 80% de ellos, la nota podrá verse incrementada un 5%. (Po a, e, k)

* Examen final: 60% (Po a, e, d, k). Este examen es obligatoria para todos los* alumnos. El alumno deberá obtener al menos una calificación de 4 (sobre 10 puntos) para poder superar la asignatura.

La nota final se obtiene sumando la nota de las pruebas parciales (según su peso). Para superar la asignatura es necesario obtener una nota final igual o superior a 5.

Si un alumno decide no seguir la evaluación continua, tendrá derecho a realizar un examen final (misma fecha y lugar que el examen ordinario). La nota obtenida en dicho examen equivale al 60% de la nota final.

En la convocatoria extraordinaria, el examen final valdrá el 100% de la nota. La evaluación continua podrá aplicarse si es más beneficio para el alumno (pruebas parciales y el 60% de la nota del examen final).

BIBLIOGRAFÍA BÁSICA

- Aho, A. V.; Hopcroft, J. E.; Ullman, J. D. Estructuras de Datos y Algoritmos, Addison Wesley Iberoamericana.
- Mark Allen Weiss Data Structures and Algorithms analysis in Java, 2nd edition, 2007, Pearson Addison Wesley.
- Michael T. Goodrich and Roberto Tamassia Data Structures and Algorithms in JAVA, 4th edition, 2006, John Wiley & Sons.

BIBLIOGRAFÍA COMPLEMENTARIA

- Isabel Segura Bedmar, Harith AlJumaily, Julian Moreno Schneider, Juan Perea & Nathan D. Ryan Algorithms and Data Structures, OCW-UC3M: <http://ocw.uc3m.es/ingenieria-informatica/algorithms-and-data-structures>, 2011

- Lourdes Araujo Serna, Raquel Martínez Unanue y Miguel Rodríguez Artacho Programación y estructuras de datos avanzadas, UNED, 2011

RECURSOS ELECTRÓNICOS BÁSICOS

- Isabel Segura Bedmar, Harith AlJumaily, Julian Moreno Schneider, Juan Perea & Nathan D. Ryan . ALGORITHMS AND DATA STRUCTURES: <http://ocw.uc3m.es/ingenieria-informatica/algorithms-and-data-structures>