Guía docente
340368 - FOPR-I1O23 - Fundamentos de Programación

Última modificación: 25/06/2024

Unidad responsable: Escuela Politécnica Superior de Ingeniería de Vilanova i la Geltrú
Unidad que imparte: 723 - CS - Departamento de Ciencias de la Computación.

Titulación: GRADO EN INGENIERÍA INFORMÁTICA (Plan 2018). (Asignatura obligatoria).

Curso: 2024 **Créditos ECTS:** 7.5 **Idiomas:** Catalán

PROFESORADO

Profesorado responsable: BERNARDINO CASAS FERNÁNDEZ

Otros: BERNARDINO CASAS FERNÁNDEZ
Alfonso Da Silva Saavedra

CAPACIDADES PREVIAS

Conocimientos básicos de matemáticas correspondientes al nivel exigido en las pruebas de acceso a la Universidad.

COMPETENCIAS DE LA TITULACIÓN A LAS QUE CONTRIBUYE LA ASIGNATURA

Específicas:
3. CEFB3. Capacidad para comprender y dominar los conceptos básicos de matemática discreta, lógica, algorítmica y complejidad computacional, y su aplicación para el tratamiento automático de la información por medio de sistemas computacionales y su aplicación para la resolución de problemas propios de la ingeniería.
4. CEFB4. Conocimiento de los fundamentos del uso y programación de los computadores, los sistemas operativos, las bases de datos y, en general, los programas informáticos con aplicación en ingeniería.
5. CEFC6. 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.
6. CEFC7. Conocimiento, diseño y utilización de forma eficiente los tipos y estructuras de datos más adecuados a la resolución de un problema.

Transversales:
1. APRENDIZAJE AUTÓNOMO: Detectar deficiencias en el propio conocimiento y superarlas mediante la reflexión crítica y la elección de la mejor actuación para ampliar este conocimiento.
2. TRABAJO EN EQUIPO: Ser capaz de trabajar como miembro de un equipo interdisciplinar ya sea como un miembro más, o realizando tareas de dirección con la finalidad de contribuir a desarrollar proyectos con pragmatismo y sentido de la responsabilidad, asumiendo compromisos teniendo en cuenta los recursos disponibles.

METODOLOGÍAS DOCENTES

La asignatura consta de:
- 2 horas a la semana de clases presenciales en el aula (grupo grande) en las que el profesor expone los contenidos.
- 3 horas a la semana en el aula de laboratorio (grupo pequeño) en las que se realizan trabajo práctico y se proponen y realizan las actividades evaluables.

OBJETIVOS DE APRENDIZAJE DE LA ASIGNATURA

1. Comprender el proceso de construcción de un programa y saber utilizar las herramientas que se requieren: consola, editor y compilador.
2. Conocer la sintaxis y la semántica de las expresiones y instrucciones básicas de un lenguaje de programación imperativo (C++).
3. Tener destreza en el uso de funciones y acciones en el desarrollo de programas.
4. Entender los conceptos de función, acción y paso de parámetros.
5. Conocer en profundidad las tablas y identificar aquellos problemas en los que su uso es apropiado.
6. Ser capaz de confrontar soluciones respecto al uso que se hacer de los recursos de tiempo y memoria y escoger la más apropiada en casos sencillos.
7. Asimilar en profundidad los esquemas de recorrido y búsqueda.
8. Asociar a un problema el esquema de solución apropiado.
9. Comprender la recursividad. Ser capaz de proponer soluciones recursivas a problemas simples.
10. Asimilar en profundidad el algoritmo de búsqueda binaria en algoritmos de ordenación de inserción, selección, mergesort y quicksort.
11. Conocer en profundidad otros algoritmos fundamentales: Hörner, producto rápido, etc.
12. Ser capaz de escribir programas de una página que sean legibles, elegantes y eficientes.

HORAS TOTALES DE DEDICACIÓN DEL ESTUDIANTADO

Tipo	Horas	Porcentaje
Horas aprendizaje autónomo	112,5	60.00
Horas grupo grande	30,0	16.00
Horas grupo pequeño	45,0	24.00

Dedicación total: 187.5 h

CONTENIDOS

Introducción

Descripción:
Ejemplos de programas
Algoritmos, lenguajes de programación y programas
Pasos en el diseño de programas

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Variables y sentencias

Descripción:
Variables, tipos de datos y expresiones
Composiciones:
- Asignación
- Entrada / Salida
- Composición Alternativa

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Iteraciones

Descripción:
Composición While
Composición For

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Tipos de datos y visibilidad

Descripción:
Tipos de datos
Conversión de tipos
Visibilidad

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Subprogramas: acciones y funciones

Descripción:
Concepto de Subprograma
Paso de parámetros
Funciones
Acciones

Dedicación: 7h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 2h

Algoritmos y secuencias. Invariantes.

Descripción:
Algoritmos y secuencias:
- Algoritmos de Recorrido
- Algoritmos de Búsqueda
Razonamiento sobre bucles: invariantes

Dedicación: 7h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 2h

Recursividad

Descripción:
Diseño recursivo
Ejemplos

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Vectores

Descripción:
Vectores
Búsqueda en vectores

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Vectores y strings

Descripción:
Más ejemplos de vectores
Strings

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo mediano/Prácticas: 3h
Aprendizaje autónomo: 1h

Vectores multidimensionales

Descripción:
Matrices
Vectores n-dimensionales
Búsqueda en una matriz
Búsqueda en una matriz ordenada
Multiplicación de matrices

Dedicación: 8h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 4h
Aprendizaje autónomo: 2h

Estructuras y diseño de estructuras de datos

Descripción:
Estructuras
Diseño de estructuras de datos

Dedicación: 6h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 1h

Ordenación

Descripción:
Ordenación por Selección
Ordenación por Inserción
Método de la Burbuja
Merge Sort

Dedicación: 8h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 4h
Aprendizaje autónomo: 2h

Algoritmos Numéricos

Descripción:
Producto de polinomios
Suma de polinomios
Suma de vectores semi vacíos
Raíces de una función continua

Dedicación: 9h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 4h
Aprendizaje autónomo: 3h

Ejemplos avanzados

Descripción:
Torneo de deportes
Permutaciones
Sub-secuencias de suma n

Dedicación: 7h
Grupo grande/Teoría: 2h
Grupo pequeño/Laboratorio: 3h
Aprendizaje autónomo: 2h

Conclusiones

Descripción:
Porqué es dificil programar?
Programas útiles
Programas correctos
Programas eficientes
Los programas son objectos matemáticos
Programas fáciles de entender, modificar y ampliar
La programación tiene límites
Citas

Dedicación: 2h
Grupo grande/Teoría: 2h

SISTEMA DE CALIFICACIÓN

Actos evaluativos obligatorios:
QU = Calificación obtenida en los cuestionarios, todos con el mismo peso.
AC = Calificación obtenida en las actividades, todas con el mismo peso.
PR = Calificación obtenido en la práctica.
C1 = Calificación obtenida en el Control 1.
C2 = Calificación obtenida en el Control 2.
PV = Prueba de Validación de la práctica.

Nota Final antes de la reevaluación = max (50%C2, 20%C1+30%C2) + 10%QU + 20%AC + 20%(PR*PV)

Acto evaluativo para la reevaluación:
C3 = Calificación obtenida en la Prueba de reavaluación

Nota Final después de la reevaluación = min (7, 50%C3 + 5%QU + 25%AC + 20%(PR*PV))

NORMAS PARA LA REALIZACIÓN DE LAS PRUEBAS.

Las actividades (AC), los controles (C1, C2 y C3) y la prueba de validación (PV) son presenciales e individuales.
Los cuestionarios (QU) son auto-evaluables, la entrega es electrónica e individual.
La práctica (PR) se realiza en grupo. El profesorado puede pedir a los estudiantes que expliquen el trabajo presentado y tener en cuenta sus respuestas a la hora de la calificación. La Prueba de Validación (PV) se realizará conjuntamente con el Control 2.