Universitat Autònoma de Barcelona

2023/2024

Fundamentos de Programación

Código: 102154
Créditos ECTS: 6

Titulación	Tipo	Curso	Semestre
2501232 Empresa y Tecnología	FB	1	2

Contacto

Nombre: Roger Malet Munté

Correo electrónico: roger.malet@uab.cat

Idiomas de los grupos

Puede consutarlo a través de este enlace. Para consultar el idioma necesitará introducir el CÓDIGO de la asignatura. Tenga en cuenta que la información es provisional hasta el 30 de noviembre del 2023.

Equipo docente

Vanessa Moreno Font

Sergi Carreras Sala

Marc Vallribera Ros

Prerrequisitos

Para poder seguir correctamente esta asignatura se necesitan unos conocimientos previos de algoritmica y resolución de problemas. Estos conocimientos se alcanzan en la asignatura "Introducción a la Resolución de Problemas y Algoritmos" (102.151), que se da el primer semestre y que da los mecanismos necesarios para saber afrontar los problemas que deberán resolverse mediante la programación en un lenguaje de alto nivel.

Objetivos y contextualización

Los objetivos de esta asignatura son los de introducir los conceptos básicos de la programación de aplicaciones informáticas para resolver problemas que se pueden dar en el ámbito empresarial. La idea de este curso es poder implementar algoritmos en un lenguaje de programación para poder generar soluciones finales que resuelvan problemas concretos y automatizar ciertos procesos informáticos que se dan en el entorno de las empresas. Además, entender el proceso de generación de aplicaciones debe permitir una mayor comprensión de los procesos informáticos más complejas que se dan en este mismo ámbito.

Así, los objetivos formativos de la asignatura se pueden resumir en:

- Entender y utilizar las estructuras básicas de la programación estructurada
- Implementar algoritmos en un lenguaje de programación de alto nivel

Competencias

- Capacidad de análisis y de síntesis.
- Demostrar la motivación por la calidad en los objetivos y en el desarrollo de su trabajo.
- Desarrollar estrategias de aprendizaje autónomo.
- Encontrar soluciones algorítmicas y utilizar las herramientas de programación adecuadas para su implementación en el entorno de una organización.
- Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.

Resultados de aprendizaje

1. Capacidad de análisis y de síntesis.
2. Codificar el algoritmo, utilizando las estructuras de programación estructurada.
3. Demostrar la motivación por la calidad en los objetivos y en el desarrollo de su trabajo.
4. Desarrollar estrategias de aprendizaje autónomo.
5. Plantear el algoritmo más eficiente para resolver el problema.
6. Proponer soluciones basadas en programación orientada a objetos.
7. Redactar de forma adecuada informes técnicos adaptados a las exigencias de sus destinatarios.
8. Utilizar las estructuras básicas de programación estructurada.

Contenido

Tema 1: Introducción

• Objetivo:

Introducir la forma en que los ordenadores procesan la información, el papel de los lenguajes de programación y el proceso de desarrollar aplicaciones.

• Contenido:

1.1. Ordenadores y algoritmos: Procesamiento de la información
1.2. Lenguajes de programación y programas
1.3. Desarrollo de software con el ordenador
1.4. Almacenamiento de la información en el ordenador

Tema 2: Iniciación a la programación en C #

• Objetivo:

Primer contacto con un lenguaje de programación, y sus características principales.

• Contenido:

2.1. Mi primer programa
2.2. Elementos del lenguaje de programación
2.3. Los datos el lenguaje de programación
2.4. Las variables
2.5. Operando con datos
2.6. La entrada / salida
2.7. Ejemplos

Tema 3: Estructuras de control

• Objetivo:

Introducción a las estructuras que permiten a un programa tomar decisiones y aplicar algoritmos básicos.

• Contenido:

3.1. Estructura general de un programa: estructuras de control
3.2. Estructuras secuencia
3.3. Estructuras de selección
3.4. Estructuras de iteración
3.5. Más sobre el operador incremento / decremento
3.6. Ejemplos

Tema 4: Programación estructurada

• Objetivo:

Introducción a las estructuras que permiten organizar el código para que las aplicaciones sean legibles y escalables.

• Contenido:

4.1. Programación estructurada y modular
4.2. Procedimientos y funciones
4.3. Funciones / Procedimientos
4.4. Variables y paso de parámetros
4.5. La recursividad
4.6. Ejemplos

Tema 5: Vectores (arrays): Listas y tablas

• Objetivo:

Estudio de los elementos que permiten crear listas y tablas de elementos en el lenguaje C #.

• Contenido:

5.1. Introducción
5.2. Vectores (listas)
5.3. Cadenas (listas) de caracteres
5.4. Matrices
5.5. Paso de parámetros
5.6. Operando con vectores (listas)
5.7. Ejemplos

Tema 6: Estructuras

• Objetivo:

Introducción a los tipos de datos definidos por el programador, que ayudan a crear tipos más complejos que los que define el propio lenguaje C.

• Contenido:

6.1. Introducción a las estructuras (registros)
6.2. Anidación de estructuras
6.3. Paso de estructuras
6.4. Ejemplos

Tema 7: Archivos

• Objetivo:

Introducción a los mecanismos que permiten tratar ficheros en C #.

• Contenido:

7.1. Introducción
7.2. Escritura de archivos
7.3. Lectura de ficheros
7.4. Funciones y opciones de lectura / escritura
7.5. Ficheros de acceso aleatorio
7.6. Paso de argumentos por el nombre de programa

Tema 8: Introducción a la programación orientada a objetos (OO)

• Objetivo:

Introducción al paradigma de programación que ve un programa como una serie de objetos o entidades que se relacionan entre ellos.

• Contenido:

8.1. ¿Qué es la programación O.O.?
8.2. Características de la programación O.O.
8.3. Clase e instancia
8.4. Ejemplos

Metodología

La metodología docente estará basada en tres tipos de actividad:

- Actividad dirigida: clases teóricas, prácticas y de análisis de problemas
- Actividad supervisada: asistencia a tutorías y realización de osecicos con seguimiento pautado
- Actividad autónoma: parte de estudio del alumno y resolución de casos, individualmente o en grupo

Debe tenerse en cuenta que la metodología docente propuesta puede experimentar alguna modificación en función de las restricciones a la presencialidad que impongan las autoridades sanitarias.

Nota: se reservarán 15 minutos de una clase dentro del calendario establecido por el centro o por la titulación para que el alumnado rellene las encuestas de evaluación de la actuación del profesorado y de evaluación de la asignatura o módulo.

Actividades

Título	Horas	ECTS	Resultados de aprendizaje
Tipo: Dirigidas			
Clases de problemas	10	0,4	2, 5, 6, 8
Clases teóricas	29,5	1,18	2, 5, 6, 8
Sesiones presenciales de laboratorio	10	0,4	2, 5, 6, 8
Tipo: Supervisadas			

Tutorias	15	0,6	2, 5, 6, 8
Tipo: Autónomas			
Preparación y resolución de ejercicios y prácticas	81,5	3,26	2, 5, 6, 8

Evaluación

Esta asignatura no prevé el sistema de evaluación única.

La evaluación del curso se realizará de forma continua, mediante exámenes, problemas relacionados con las clases de teoría y trabajos específicos de las sesiones de laboratorio. El peso de cada uno de los componentes será:

- 30% para el conjunto de las notas de prácticas,
- 10% los ejercicios propuestos en clase individualmente o por grupos,
- 25% el primer examen, que se efectuará a mediados del semestre, y
- 35% en el segundo examen, que se efectuará a finales de semestre, en la fecha fijada por la facultad. Quien haya obtenido en el primer examen una calificación mínima de 5, pueden optar por realizar una prueba sustitutoria adicional en el aula de informática, en la fecha que se fijará oportunamente, y que coincidirá con la semana final de clases.

Se superará la asignatura si se obtiene una nota mínima de 5.0 en la media ponderada de las notas obtenidas en las actividades descritas anteriormente. Para poder efectuar dicha media, deberán cumplirse las siguientes condiciones:

- Se deberá haber entregado todas las prácticas, y haber obtenido una nota mínima de 5.
- La media ponderada de los exámenes debe superar el 4.

Si en la evaluación final se ha obtenido una nota superior a 3,5 y no se llega al 5, tendrán derecho a una recuperación, en los términos que se describen más abajo.

La asignatura se considerará "no evaluable" cuando no haya participado en ninguna de las actividades de evaluación.

Calendario de actividades de evaluación

Las fechas de las diferentes actividades de evaluación (ejercicios, entrega de trabajos, etc.) se anunciarán con suficiente antelación durante elsemestre.

Las fechas de los exámenes parcial y final de la asignaturaestan programadas en el calendario de exámenes de la Facultad.

"*La programación de las pruebas de evaluación no se podrá modificar, salvo que haya un motivo excepcional y debidamente justificado por el cual no se pueda realizar un acto de evaluación. En este caso, las personas responsables de las titulaciones, previa consulta al profesorado y al estudiantado afectado, propondrán una nueva programación dentro del período lectivo correspondiente*." ***Apartado 1 del Artículo 115. Calendario de las actividades de evaluación (Normativa Académica UAB)***

Los y las estudiantes de la Facultad de Economía y Empresa que, de acuerdo con el párrafo anterior, necesiten cambiar una fecha de evaluación deben presentar la petición rellenando el documento Solicitud reprogramación prueba en https://eformularis.uab.cat/group/deganat_feie/solicitud-reprogramacion-de-pruebas

Procedimiento de revisión de las calificaciones

Coincidiendo con el examen final se anunciará el día y el medio de publicación de las calificaciones finales. Asimismo se informará del procedimiento, lugar, fecha y hora de la revisión de las mismas de acuerdo con la normativa de la Universidad.

Proceso de Recuperación

"Para participar en el proceso de recuperación el alumnado debe haber sido previamente evaluado en un conjunto de actividades que represente un mínimo de dos terceras partes de la calificación total de la asignatura o módulo." ***Apartado 3 del Artículo 112 ter. La recuperación (Normativa Académica UAB).*** Los y las estudiantes deben haber obtenido una calificación media de la asignatura entre 3,5 y 4,9.

La fecha de esta prueba está programada en el calendario de exámenes de la Facultad. El estudiante que se presente y la supere aprobará la asignatura con una nota de 5. En caso contrario mantendrá la misma nota.

Irregularidades en actos de evaluación

Sin perjuicio de otras medidas disciplinarias que se estimen oportunas, y de acuerdo con la normativa académica vigente, *"en caso que el estudiante realice cualquier irregularidad que pueda conducir a una variación significativa de la calificación de un acto de evaluación, se calificará con un 0 este acto de evaluación, con independencia del proceso disciplinario que se pueda instruir. En caso que se produzcan diversas irregularidades en los actos de evaluación de una misma asignatura, la calificación final de esta asignatura será 0".* ***Apartado 10 del Artículo 116. Resultados de la evaluación. (Normativa Académica UAB)***

Nota final: *La evaluación propuesta puede experimentar alguna modificación en función de las restricciones a la presencialidad que impongan las autoridades sanitarias.*

Actividades de evaluación continuada

Título	Peso	Horas	ECTS	Resultados de aprendizaje
Ejercicios propuestos en clase	10%	0	0	2, 5, 6, 8
Entregas de las sesiones de prácticas	30%	0	0	2, 3, 4, 5, 6, 7, 1, 8
Primer examen parcial	25%	2	0,08	2, 5, 6, 8
Segundo examen parcial	35%	2	0,08	2, 4, 5, 6, 1, 8

Bibliografía

Python Tutorial - Tutorialspoint

Tutorial de Python (mundogeek.net)

A Beginners guide to Python 3 programming / John Hunt

Software

Python 3

Jupyter

Visual Studio Code