

GUÍA DOCENTE

Curso Académico 2024/25

1. Programación II

1.1. Datos de la asignatura

Tipo de estudios	Grado
Titulación	Grado en Ingeniería del Software
Nombre de la asignatura	Programación II
Carácter de la asignatura	Formación Básica
Curso	1
Idioma de impartición	Español
Coordinador/a de la asignatura	Peralta Samaniego, Federico Daniel
Semestre	2º Semestre
Número de créditos	6

Course data

Level of study	Degree
Degree	Bachelor's Degree in Software Engineering
Course name	Computer Programming II
Course character	Fundamental Courses
Year of study	1
Language	Spanish
Course coordinator	Peralta Samaniego, Federico Daniel
Semester	2nd Semester

Córdoba/Sevilla/Granada, 9 de julio de 2024

Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100
Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600
Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

ECTS Credits	6
--------------	---

1.2. Datos del equipo de profesores

Escuela Técnica Superior de Ingeniería
 Página web: www.uloynola.es
 Departamento de Métodos Cuantitativos

Apellidos, Nombre	Correo electrónico
Parody Núñez, María Luisa	mlparody@uloynola.es
Peralta Samaniego, Federico Daniel	fdperalta@uloynola.es
Maradiaga Chirinos, Jorge Raúl	jrmaradiaga@uloynola.es
Gantiva Osorio, Manuel Eduardo	megantiva@uloynola.es

1.3. Requisitos previos

Haber cursado la asignatura Fundamentos de Informática I

1.4 Objetivos del curso

Conocer el paradigma de la programación orientada a objetos, realizar un diagrama de clases para resolver un problema específico, empleando las propiedades de los sistemas orientado a objetos, saber programar y depurar una aplicación modular orientada a objetos, con especial énfasis en el tratamiento de objetos matemáticos y datos.

Competencias básicas:

- CB1 - Que los estudiantes hayan demostrado poseer y comprender conocimientos en un área de estudio que parte de la base de la educación secundaria general, y se suele encontrar a un nivel que, si bien se apoya en libros de texto avanzados, incluye también algunos aspectos que implican conocimientos procedentes de la vanguardia de su campo de estudio.
- CB2 - Que los estudiantes sepan aplicar sus conocimientos a su trabajo o vocación de una forma profesional y posean las competencias que suelen demostrarse por medio de la elaboración y defensa de argumentos y la resolución de problemas dentro de su área de estudio.
- CB3 - Que los estudiantes tengan la capacidad de reunir e interpretar datos relevantes (normalmente dentro de su área de estudio) para emitir juicios que incluyan una reflexión sobre temas relevantes de índole social, científica o ética.
- CB4 - Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
- CB5 - Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía.

Competencias generales:

- CT2 - Conocimiento en materias básicas y tecnológicas, que les capacite para el aprendizaje de nuevos métodos y teorías, y les dote de versatilidad para adaptarse a nuevas situaciones.

Córdoba/Sevilla/Granada, 9 de julio de 2024
 Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100
 Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600
 Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

- CT3 - Capacidad de resolver problemas con iniciativa, toma de decisiones, creatividad, razonamiento crítico y de comunicar y transmitir conocimientos, habilidades y destrezas en el campo de la Ingeniería Informática.

Competencias específicas:

- CFB1 - Capacidad para la resolución de los problemas matemáticos que puedan plantearse en la ingeniería. Aptitud para aplicar los conocimientos sobre: álgebra lineal; cálculo diferencial e integral; métodos numéricos; algorítmica numérica; estadística y optimización.
- CFB4 - Conocimientos básicos sobre el uso y programación de los ordenadores, sistemas operativos, bases de datos y programas informáticos con aplicación en ingeniería.
- CFB5 - Conocimiento de la estructura, organización, funcionamiento e interconexión de los sistemas informáticos, los fundamentos de su programación, y su aplicación para la resolución de problemas propios de la ingeniería.

1.5. Objetivos de desarrollo sostenible (ODS)

4. Garantizar una educación inclusiva, equitativa y de calidad y promover oportunidades de aprendizaje durante toda la vida para todos

4.4. Competencias para acceder al empleo.

1.6. Asistencia a clase

Dado que la asignatura posee un fuerte componente práctico, con evaluación continua del seguimiento del trabajo del estudiante, se da una importancia especial a la asistencia. La participación activa del estudiante, así como su evolución a lo largo del mismo, será supervisada por el profesor, en forma de cuestiones en clase, resolución de problemas, tareas diseño o programación, etc.

El sistema de evaluación continua requiere que el estudiante asista un mínimo de un 70% de las actividades de carácter presencial para poder presentarse tanto a Convocatoria Ordinaria como Extraordinaria. Cada profesor deberá realizar el control de asistencia a los alumnos que imparte docencia en la forma objetiva que considere oportuna. Por causa de fuerza mayor debidamente justificada ante el Director de la titulación, podrá eximirse del mínimo de asistencia señalado.

1.7. Contenidos del programa

1. **Punteros, tipos de datos complejos y gestión de la memoria**
 - Funciones
 - Punteros y memoria dinámica
 - Tipos de datos de texto
 - Introducción biblioteca estándar
 - Arrays y matrices
 - Estructuras
2. **Programación modular e introducción a la Orientación a Objetos**
 - Programación con clases y objetos.
 - Conceptos de abstracción, encapsulación, herencia y polimorfismo.
 - Objetos, referencias y punteros
 - Organización del código y bibliotecas
3. **Flujos de entrada/salida:**
 - Flujos E/S.
 - Ficheros de texto y binarios
 - Interacción con el sistema operativo

Córdoba/Sevilla/Granada, 9 de julio de 2024
Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100
Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600
Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

4. **Introducción a estructuras de datos**
 - Iteradores
 - Vectores
 - Listas
 - Colas
 - Pilas
5. **Introducción a algoritmos de búsqueda y ordenación**
 - Algoritmos de búsqueda
 - Algoritmos de ordenación de vectores

Contents

1. **Pointers, complex types and memory management**
 - Functions in C++
 - Pointers and dynamic memory
 - Text data types
 - Introduction to the C++ standard library
 - Arrays and matrices
 - Structures
2. **Modular programming and introduction to object oriented programming**
 - Programming with classes and objects.
 - Concepts of abstraction, encapsulation, inheritance and polymorphism
 - Objects, references and pointers
 - Code design and libraries.
3. **Input/output streams:**
 - I/O streams.
 - Files
 - Operating system interaction
4. **Introduction to structures in C++:**
 - Iterators
 - Vectors
 - List
 - Queues
 - Stacks
5. **Search and sorting algorithms:**
 - Search algorithms
 - Sorting algorithms for vectors

1.8. Bibliografía básica

Título	Autor	Editorial	Año	Observaciones
COMO PROGRAMAR EN C/ C++	H. M. Deitel, Paul. J Deitel	Pearson	2011	

1.9. Bibliografía complementaria

- Referencia a C++: <https://en.cppreference.com/w/>
- Cplusplus.com: <http://www.cplusplus.com/doc/tutorial/>
- LearnCpp.com: <https://www.learncpp.com/>
- The C++ Resources Network: <https://www.cplusplus.com/info/>
- Bruce Eckel. *Pensar en C++*. 2010. Disponible online: <http://arco.esi.uclm.es/~david.villa/pensarC++.html>
- Bjarne Stroustrup. *The C++ Programming Language*. Addison Wesley, 2013. ISBN: 978-0275967307.
- Stanley B. Lippman, Josée Lajoie, y Barbara E. Moo. *C++ Primer*. Addison-Wesley Educational Publishers Inc, 2012. ISBN: 978-0321714114

Córdoba/Sevilla/Granada, 9 de julio de 2024
Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100
Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600
Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

- Gaddis T., Walters J, Muganda G. *Starting Out with C++: Early Objects (8th Edition)*. Addison Wesley 2013
- Luis Joyanes Aguilar. *Fundamentos de Programación. Algoritmos, estructuras de datos y objetos (4ª Edición)*. S.A. MCGRAW-HILL / INTERAMERICANA DE ESPAÑA, 2008.

2. Métodos Docentes

En la asignatura se emplearán fundamentalmente los siguientes métodos didácticos-docentes:

- Clases teóricas. Las clases teóricas introducirán los conceptos fundamentales de la asignatura.
- Seminarios de problemas en el aula (profesor). Al término de cada tema o apartado se resolverán en clase una serie de ejercicios ilustrativos que introduzcan al alumno en la aplicación práctica de los conceptos.
- Resolución de cuestiones y problemas por parte de los alumnos en el aula. Estas sesiones tendrán diferentes características, que van desde la respuesta telemática de cuestiones tipo test en el mismo aula para verificar la correcta asimilación de los conceptos de base, hasta la resolución de problemas de programación individuales.
- Resolución de problemas de programación modular en grupos, en los que la tarea de desarrollo se distribuye por módulos o funciones de programa (Aprendizaje Basado en Proyectos).
- Clases prácticas en el aula de ordenadores. Los alumnos se enfrentarán a series de ejercicios de dificultad creciente.
- Tutorías no programadas individuales o en grupo.

Se pondrá a disposición de los alumnos el siguiente material básico:

- Programa de la asignatura.
- Documento con los contenidos teóricos y resultados más importantes de cada tema que se van a desarrollar en clase, los ejercicios que se resolverán y los ejercicios que se proponen para el trabajo autónomo. Incluyendo las diapositivas empleadas en el aula.
- Boletín de ejercicios propuestos de cada tema para resolver en clase o para trabajo personal.
- Compiladores y software necesario para el trabajo autónomo.

2.1. Uso de la inteligencia artificial

Los ejercicios de programación en clase y en casa se realizarán promoviendo la utilización ética de generadores inteligentes de texto. El profesor tomará los cuidados para crear ejercicios que sean sobre escenarios actuales, provocadores de pensamiento y reflexión y de finales abiertos, tal que las resoluciones de los ejercicios dependan de la reflexión y el aprendizaje de los alumnos. Además, cualquier intento por parte de los alumnos de apropiarse de trabajos enteramente creados por inteligencia artificial invalidará el trabajo. Esto se logrará mediante evaluaciones orales, análisis grupales e implementaciones de *pair programming*, acciones que formentarán la integridad académica.

3. Tiempo estimado de trabajo del estudiante

1. Clases presenciales en aula: 60 h
2. Trabajo autónomo (estudio individual, desarrollo en entorno de programación de ejercicios de los boletines, trabajo de programación): 90 h

4. Métodos de evaluación y porcentaje en la calificación final

Sistema de Evaluación: Evaluación Continua

Córdoba/Sevilla/Granada, 9 de julio de 2024
Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100
Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600
Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

La evaluación de la asignatura se realizará a partir de los siguientes elementos:

- Examen final con preguntas teóricas y ejercicios
- Cuestionarios telemáticos en clase y en casa
- Programación en aula de ordenadores y validación

La ponderación de estos elementos será la siguiente:

	ELEMENTOS DE EVALUACIÓN	PUNTUACIÓN MÁXIMA	PUNTUACIÓN MÍNIMA
Cuestiones de clase 20%	Cuestionarios de clase	1 PUNTO	
	Ejercicios de Programación en aula de ordenadores y trabajo en casa	1 PUNTO	
Pruebas y ejercicios 60%	Prueba de bloque 1	3 PUNTOS	3 PUNTOS
	Prueba de bloque 2	3 PUNTOS	
Examen final 20%	Examen con preguntas teóricas y prácticas	2 PUNTOS	1 PUNTO
	Total	10 PUNTOS	5 PUNTOS

El alumno realizará *Cuestiones de clase* (20% de la nota final de la asignatura) para demostrar que los conocimientos teóricos-prácticos están siendo asimilados a lo largo de la evaluación continua. Estos se componen de *Cuestionarios de clase* (10%) compuestos por los contenidos teóricos y *Ejercicios de programación* (10%) que corresponde a la validación de ejercicios de programación a la finalización de cada tema para asimilar y aplicar los conceptos teóricos. La validación implica que los alumnos realizarán una serie de ejercicios para preparar un pequeño supuesto práctico que tendrán que resolver en clase. *Pruebas y ejercicios* (60%): *Pruebas parciales* (30%) que se corresponderán con contenidos prácticos con el objetivo de preparar el examen final de la asignatura. Cada prueba corresponde a la resolución de un problema de mayor complejidad que unifique los conceptos vistos a lo largo de la asignatura para que el alumno demuestre que se han alcanzado los objetivos de la materia. El *Examen final* (20%) supondrá una serie de ejercicios de dificultad similar a los entregados previamente. Para hacer media, la nota en cada elemento evaluable debe ser igual o superior al 50%, en caso contrario, el alumno deberá presentarse a la convocatoria extraordinaria para aprobar la asignatura. En todo caso, el profesor podrá citar al alumno para que demuestre la calificación obtenida si se producen resultados sospechosos como la no presentación de trabajos y el aprobado del examen.

Sistema de Evaluación: Convocatoria extraordinaria

Los alumnos que no superen algunas de las partes indicadas previamente deberán presentarse a la convocatoria extraordinaria para su recuperación, es decir, las partes aprobadas en la convocatoria ordinaria serán guardadas para esta convocatoria, a no ser que el alumno se presente a subir nota. El profesor enviará una relación de ejercicios para recuperar la parte de *Pruebas y ejercicios* (60%) que será diferente a la correspondiente de la evaluación continua. El *Examen final* de la convocatoria extraordinaria tendrá las mismas características que el anterior y por tanto, supondrá un 20% de la nota final.

Córdoba/Sevilla/Granada, 9 de julio de 2024

Universidad Loyola Andalucía

Campus Córdoba, C/ Escritor Castilla Aguayo 4, 14004 Córdoba (España) Tel.: (+34) 957 222 100

Campus Sevilla, Avda. de las Universidades s/n. Dos Hermanas, Sevilla (España) Tel.: (+34) 955 641 600

Campus Granada, C/ Profesor Vicente Callao, 15, 18011 Granada (España) Tel.: (+34) 958 185 252

5. Código ético

Todos los miembros de la Universidad Loyola Andalucía tendrán un comportamiento académico ético y se abstendrán conductas no éticas en todas sus formas, incluido el plagio, el engaño, la tergiversación y la falsedad. El plagio o el engaño del estudiante en el trabajo académico individual o grupal o en la conducta de examen resultará sancionada según se indica en la Normativa General Académica de la Universidad

All members of the Universidad Loyola Andalucía shall adhere to high standards of academic ethics and shall refrain from academic dishonesty and misconduct in all forms, including plagiarism, cheating, misrepresentation, fabrication, and falsehood. Plagiarism or cheating by a student in individual or group work or in examinations will be sanctioned as indicated by the General Academic Regulation of the University.