



## **GUÍA DOCENTE**

# **INTRODUCCIÓN A LA PROGRAMACIÓN I**

## **GRADO EN INGENIERÍA DEL SOFTWARE**

***MODALIDAD: A DISTANCIA***

***CURSO ACADÉMICO: 2023-2024***

Denominación de la asignatura:	<b>Introducción a la programación I</b>
Titulación:	Ingeniería del Software
Facultad o Centro:	Centro Universitario de Tecnología y Arte Digital
Materia:	Fundamentos Informáticos
Curso:	1º
Cuatrimestre:	1
Carácter:	B
Créditos ECTS:	6
Modalidad de enseñanza:	A distancia
Idioma:	Castellano
Profesor / Email:	Ana Isabel Sierra De Las Heras / ana.sierra@u-tad.com Rodrigo Alonso Solaguren-Beascoa / rodrigo.alonso@u-tad.com Marcos Novalbos Mendiguchía / marcos.novalbos@u-tad.com Alfonso Castro Escudero / alfonso.castro@u-tad.com
Página Web:	<a href="http://www.u-tad.com/">http://www.u-tad.com/</a>

## DESCRIPCIÓN DE LA ASIGNATURA

### Descripción de la materia

Esta asignatura pertenece a la materia de Fundamentos informáticos, donde se pretende que el estudiante adquiera conocimientos básicos en los que se fundamenta la ingeniería del software

### Descripción de la asignatura

La asignatura de introducción a la programación I es el primer punto de contacto de los alumnos con la creación estructurada de programas. Para ello se ha elegido C como lenguaje de programación, debido a su eficiencia a la hora de crear código con lo que el alumno puede ser capaz de entender las acciones que se desencadenan al escribir una línea de código. Se revisarán los conceptos básicos de la programación como son los tipos de datos, operadores y expresiones y el modularidad

## COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

## Competencias (genéricas, específicas y transversales)

### COMPETENCIAS BÁSICAS Y GENERALES

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

CG1 - Capacidad para entender, planificar y resolver problemas a través del desarrollo de soluciones informáticas.

CG3 - Conocimiento de los fundamentos científicos aplicables a la resolución de problemas informáticos

CG5 - Gestión de los recursos humanos y tecnológicos para la correcta realización de proyectos informáticos

CG9 - Capacidad para aprender, modificar y producir nuevas tecnologías informáticas

### COMPETENCIAS ESPECÍFICAS

CE3 - Conocimiento del álgebra relacional y realización de consultas en lenguajes procedurales para el diseño de esquemas de

bases de datos normalizados basados en modelos de entidad-relación

CE9 - Conocimiento de las estructuras de control, variables, sintaxis de programación y gestión del uso de la memoria de manera

eficaz en el desarrollo de una aplicación informática

CE11 - Conocimiento de la arquitectura de los Sistemas Operativos así como los distintos mecanismos para la gestión de procesos,

comunicación y sincronización de los mismos

CE13 - Conocimiento de los fundamentos de las redes de ordenadores, de las distintas topologías y de sus protocolos de

comunicación

CE16 - Conocimiento del funcionamiento de los sistemas informáticos

### Resultados de aprendizaje

Al acabar la titulación, el graduado o graduada será capaz de:

- Comprender el proceso de desarrollo de un programa informático en distintos lenguajes de programación.
- Saber depurar un programa software.
- Conocer y manejar los lenguajes de marcas (HTML)
- Construir páginas web usando hojas de estilos CSS
- Manejar la herramienta de control de versiones Git para el desarrollo colaborativo
- Comprender los fundamentos del álgebra de Boole
- Manejar de puertas lógicas y circuitos secuenciales sencillos
- Manejar la representación binaria de diferentes tipos de datos
- Entender el modelo de Von Neumann
- Entender la arquitectura hardware del ordenador
- Ser capaz de construir programas básicos en ensamblador
- Conocer las tecnologías de red más habituales (WiFi, BlueTooth, Ethernet¿)
- Conocer las topologías de red
- Entender cómo se comunican los ordenadores usando protocolos como ARP, IP, TCP, etc.
- Aprender a establecer configuraciones básicas de enrutamiento
- Ser capaz de construir aplicaciones de red sencillas
- Conocer la arquitectura básica de un sistema operativo
- Entender los principios de la planificación de procesos.
- Entender el funcionamiento de la jerarquía de memoria.
- Desarrollar un sistema de ficheros sencillo.
- Desarrollar un driver básico.
- Entender los mecanismos de sincronización y comunicación entre procesos y threads

### CONTENIDO

Sintaxis de programación

Uso de bibliotecas estándar

Tipos de datos, funciones y modularidad

## TEMARIO

Unidad 1: Fundamentos de Programación. Programación en pseudocódigo

Unidad 2: El lenguaje C. Elementos básicos

Unidad 3: Operadores, expresiones, estructuras de selección y de control

Unidad 4: Funciones

Unidad 5: Arrays, Cadenas de caracteres y Matrices

Unidad 6: Punteros

## ACTIVIDADES FORMATIVAS Y METODOLOGÍAS DE APRENDIZAJE

### Actividades formativas

Actividad Formativa	Horas totales	Horas síncronas
<i>Sesiones teóricas virtuales síncronas</i>	3,33	3
<i>Sesiones teóricas virtuales asíncronas</i>	25,33	0
<i>Sesiones prácticas virtuales síncronas</i>	2,67	3
<i>Sesiones prácticas virtuales asíncronas</i>	12,67	0
<i>Debate y discusión oral y/o escrita.</i>	8,00	0
<i>Tutorías</i>	4,00	4
<i>Estudio independiente y trabajo autónomo del alumno</i>	49,33	0
<i>Elaboración de trabajos (en grupo o individuales)</i>	34,67	0
<i>Actividades de Evaluación</i>	4,00	4
<i>Test de autoevaluación</i>	6,00	0
<i>Prácticas externas</i>	0,00	0
<i>Preparación y defensa virtual del TFG</i>	0,00	0
<i>Seguimiento de proyectos</i>	0,00	0
<b>TOTAL</b>	<b>150</b>	<b>14</b>

### Metodologías docentes

Método expositivo o lección magistral

Aprendizaje de casos

Aprendizaje basado en la resolución de problemas

Aprendizaje cooperativo o colaborativo

Aprendizaje por indagación

Metodología Flipped classroom o aula invertida

Gamificación

Just in time Teaching (JITT) o aula a tiempo

Método expositivo o lección magistral

Método del caso

Aprendizaje basado en la resolución de problemas

Aprendizaje cooperativo o colaborativo

Aprendizaje por indagación

Metodología flipped classroom o aula invertida

Gamificación

### DESARROLLO TEMPORAL

Presentación - semana 1

Unidad 1 - semana 2-3

Unidad 2 - semana 4-5

Unidad 3 - semana 6-7

Unidad 4 - semana 7-8

Unidad 5 - semana 9-10

Unidad 6 - semana 11-12

Repaso - semana 13-14

Evaluación - semana 15

### SISTEMA DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	VALORACIÓN MÍNIMA RESPECTO A LA CALIFICACIÓN FINAL (%)	VALORACIÓN MÁXIMA RESPECTO A LA CALIFICACIÓN FINAL (%)
-------------------------	--	--

<i>Evaluación de la participación en clase, en prácticas o en proyectos de la asignatura</i>	10	20
<i>Evaluación de trabajos, proyectos, informes, memorias</i>	10	20
<i>Prueba Objetiva</i>	60	70

## CRITERIOS ESPECÍFICOS DE EVALUACIÓN

ACTIVIDAD DE EVALUACIÓN	CONVOCATORIA ORDINARIA	CONVOCATORIA EXTRAORDINARIA
<i>Evaluación de la participación en clase, en prácticas o en proyectos de la asignatura</i>	20	10
<i>Evaluación de trabajos, proyectos, informes, memorias</i>	20	20
<i>Prueba Objetiva</i>	60	70

### Consideraciones específicas acerca de la evaluación

Será necesario que obtener una nota mínima de 4 puntos (sobre 10) en la prueba final presencial para que se realice la media con las actividades formativas.

## BIBLIOGRAFÍA / WEBGRAFÍA

Bibliografía básica:

David Griffiths, Dawn Griffiths, Head First C, O'Reilly Media, ISBN: 978-1-4493-9991-7.

Luis Joyanes, Fundamentos de programación. McGraw-Hill Interamericana de España S.L. ISBN-13: 978-8448161118.

Bibliografía recomendada:

Brian W. Kernighan, Dennis M. Ritchie, El lenguaje de Programación C. Prentice Hall, ISBN 9789688802052.

Fundamentos de programación utilizando el lenguaje C. Universidad Pontificia de Comillas. ISBN: 978-84-8468-184-7.

"C Programming Absolute beginners' Guide". ISBN: 978-0789751980.

Martín Coin, "Caminando junto al lenguaje C" - Editorial UNRN.

Bibliografía recomendada:

Introducción a Computer Science, incluido en la página del profesor Bourke:  
<https://bitbucket.org/chrisbourke/computersciencei/src/master/ComputerScienceOne.pdf>.

<https://youtu.be/CNFK86hJRfE>

<https://www.youtube.com/watch?v=ssJY5MDLjlo>.

[https://ocw.mit.edu/courses/6-s096-introduction-to-c-and-c-january-iap-2013/resources/mit6\\_s096\\_iap13\\_lec1/](https://ocw.mit.edu/courses/6-s096-introduction-to-c-and-c-january-iap-2013/resources/mit6_s096_iap13_lec1/)

## **MATERIALES, SOFTWARE Y HERRAMIENTAS NECESARIAS**

### **Materiales:**

Material básico para cursar la asignatura:

Ordenador personal

Webcam

Micrófono

Material específico de la asignatura:

Cygwin: Emulador de Linux en Windows

Editor de texto: Notepad ++

Compilador C: gcc

### **Software:**

Cygwin: Emulador de Linux en Windows

Editor de texto: Notepad ++

Compilador C: gcc