

## Programación semanal

En la programación semanal te presentamos un **reparto del trabajo de la asignatura** a lo largo de las semanas del cuatrimestre.

	Temas	Actividades (7,5 puntos)	Eventos (7,5 puntos)
Semana 1	<b>Tema 1. Conceptos básicos sobre programación</b> 1.1. ¿Cómo estudiar este tema? 1.2. Concepto de computador 1.3. Los lenguajes de programación 1.4. Ensambladores, compiladores e intérpretes		Asistencia a 2 sesiones presenciales virtuales (0,45 puntos cada una)  <b>Test Tema 1</b> (0,3 puntos)
Semana 2	<b>Tema 2. Programación modular y estructurada</b> 2.1. ¿Cómo estudiar este tema? 2.2. Estructura de un programa 2.3. Algoritmos 2.4. Programación modular y estructurada		<b>Test Tema 2</b> (0,3 puntos)
Semana 3	<b>Tema 3. Tipos de datos</b> 3.1. ¿Cómo estudiar este tema? 3.2. Constantes y variables 3.3. Expresiones, identificadores y palabras reservadas 3.4. Tipos de datos simples: numéricos, caracteres, lógicos 3.5. Tipos de datos compuestos: cadenas, vectores, matrices y listas		<b>Test Tema 3</b> (0,3 puntos)  <b>Foro de debate:</b> Propuestas para el diseño de un programa con cadenas de caracteres en pseudocódigo (3 puntos)
Semana 4	<b>Tema 4. Estructuras de control</b> 4.1. ¿Cómo estudiar este tema? 4.2. Tipos de estructuras de control 4.3. Estructura secuencial 4.4. Estructuras selectivas: bifurcación condicional		
Semana 5	<b>Tema 4. Estructuras de control</b> 4.5. Estructuras repetitivas: bucles 4.6. Bifurcaciones incondicionales	<b>Trabajo:</b> Diseño de un algoritmo (1,2 puntos)	<b>Test Tema 4</b> (0,3 puntos)
Semana 6	<b>Tema 5. Introducción a la programación en Python</b> 5.1. ¿Cómo estudiar este tema? 5.2. Estructura de un programa 5.3 Módulos y paquetes		
Semana 7	<b>Tema 5. Introducción a la programación en Python</b> 5.3 Módulos y paquetes 5.4. Tipos de datos simples 5.5. Estructuras de control	<b>Trabajo:</b> Escribir tres algoritmos (1,2 puntos)	<b>Test Tema 5</b> (0,3 puntos)
Semana 8	<b>Tema 6. Subprogramas</b> 6.1. ¿Cómo estudiar este tema? 6.2. Introducción a los subprogramas 6.3. Funciones y procedimientos 6.4. Ámbito y visibilidad 6.5. Funciones en Python	<b>Trabajo:</b> Copiar una imagen (2,5 puntos)	<b>Test Tema 6</b> (0,3 puntos)

	Temas	Actividades (7,5 puntos)	Eventos (7,5 puntos)
Semana 9	<b>Tema 7. Estructuras de datos</b> 7.1. ¿Cómo estudiar este tema? 7.2. Qué es la programación orientada a objetos 7.3. Conceptos fundamentales 7.4. POO en Python		<b>Test Tema 7</b> (0,3 puntos)
Semana 10	<b>Tema 8. Funciones en Python</b> 8.1. ¿Cómo estudiar este tema? 8.2. Introducción 8.3. Listas 8.4. Tuplas 8.5. Diccionarios 8.6. Pilas 8.7. Colas	<b>Trabajo:</b> Método para pilas y colas (1,3 puntos)	<b>Test Tema 8</b> (0,3 puntos)
Semana 11	<b>Tema 9. Archivos y ficheros</b> 9.1. ¿Cómo estudiar este tema? 9.2. Conceptos y definiciones 9.3. Soporte y acceso 9.4. Clasificación de archivos y operaciones 9.5. Procesamiento de archivos		<b>Test Tema 9</b> (0,3 puntos)
Semana 12	<b>Tema 10. Recursividad</b> 10.1. ¿Cómo estudiar este tema? 10.2. Recursividad: definición 10.3. Diseño de algoritmos recursivos 10.4. Recursión e iteración		<b>Test Tema 10</b> (0,3 puntos)
Semana 13	<b>Tema 10. Recursividad</b> 10.5. Ejemplos		
Semana 14	<b>Tema 11. Ficheros y recursividad en Python</b> 11.1. ¿Cómo estudiar este tema? 11.2. Ficheros 11.3. Recursividad	<b>Trabajo:</b> Funciones de Python (1,3 puntos)	<b>Test Tema 11</b> (0,3 puntos)
Semana 15	<b>Tema 12. Documentación y depuración</b> 12.1. ¿Cómo estudiar este tema? 12.2 Documentación de programas 12.3 Documentación de especificaciones 12.4 Guía técnica 12.5 Manual de usuario 12.6 Guía de instalación 12.7 Depuración de programas		<b>Test Tema 12</b> (0,3 puntos)
Semana 16	<b>Semana de exámenes</b>		