

Programación semanal

Para conocer la fecha de entrega de las distintas actividades accede a la sección **Actividades** (en el menú lateral). Recuerda que la suma de las puntuaciones de todas las actividades es de 15 puntos. Puedes hacer las que prefieras hasta conseguir un máximo de 10 puntos (que es la calificación máxima que se puede obtener en la evaluación continua).

	Contenido teórico	Actividades (12.0 puntos)	Eventos (3.0 puntos)
Semana 1	Tema 1. Conceptos básicos sobre programación 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 a lo largo de la asignatura (0,2 puntos cada una) Test - Tema 01 (0.1 puntos)
Semana 2	Tema 2. Programación modular y estructurada 2.1. ¿Cómo estudiar este tema? 2.2. Estructura de un programa 2.3. Algoritmos 2.4. Programación modular y estructurada		Test - Tema 02 (0.1 puntos)
Semana 3	Tema 3. Tipos de datos 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		Foro: Propuestas para el diseño de un programa con cadenas de caracteres en pseudocódigo (1.4 puntos) Test - Tema 03 (0.1 puntos)
Semana 4	Tema 4. Estructuras de control 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	Tema 4. Estructuras de control 4.5. Estructuras repetitivas: bucles 4.6. Bifurcaciones incondicionales	Trabajo: Diseño de un algoritmo (2.0 puntos)	Test - Tema 04 (0.1 puntos)
Semana 6	Tema 5. Introducción a la programación en Python 5.1. ¿Cómo estudiar este tema? 5.2. Estructura de un programa 5.3 Módulos y paquetes		
Semana 7	Tema 5. Introducción a la programación en Python 5.3 Módulos y paquetes 5.4. Tipos de datos simples 5.5. Estructuras de control	Trabajo: Escribir dos algoritmos (2.0 puntos)	Test - Tema 05 (0.1 puntos)

	Contenido teórico	Actividades (12.0 puntos)	Eventos (3.0 puntos)
Semana 8	Tema 6. Subprogramas 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	Trabajo: Hacer un dibujo (4.0 puntos)	Test - Tema 06 (0.1 puntos)
Semana 9	Tema 7. Estructuras de datos 7.1. ¿Cómo estudiar este tema? 7.2. Qué es la programación orientada a objetos 7.3. Conceptos fundamentales 7.4. POO en Python		Test - Tema 07 (0.1 puntos)
Semana 10	Tema 8. Funciones en Python 8.1. ¿Cómo estudiar este tema? 8.2. Introducción 8.3. Listas 8.4. Tuplas 8.5. Diccionarios 8.6. Pilas 8.7. Colas	Trabajo: Método para pilas y colas (2.0 puntos)	Test - Tema 08 (0.1 puntos)
Semana 11	Tema 9. Archivos y ficheros 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		Test - Tema 09 (0.1 puntos)
Semana 12	Tema 10. Recursividad 10.1. ¿Cómo estudiar este tema? 10.2. Recursividad: definición 10.3. Diseño de algoritmos recursivos 10.4. Recursión e iteración		Test - Tema 10 (0.1 puntos)
Semana 13	Tema 10. Recursividad 10.5. Ejemplos	Trabajo: Funciones en Python (2.0 puntos)	
Semana 14	Tema 11. Ficheros y recursividad en Python 11.1. ¿Cómo estudiar este tema? 11.2. Ficheros 11.3. Recursividad		Test - Tema 11 (0.1 puntos)
Semana 15	Tema 12. Documentación y depuración 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		Test - Tema 12 (0.1 puntos)
Semana 16	Semana de exámenes		