



Fundamentos de
programación
**Grado en Matemáticas
Aplicadas**



UNIVERSIDAD
NEBRIJA

GUÍA DOCENTE

Asignatura: Fundamentos de programación

Titulación: Grado en Matemáticas Aplicadas

Carácter: Básica

Idioma: Castellano

Modalidad: Presencial

Créditos: 6

Curso: 1º

Semestre: 2º

Profesores/Equipo docente: Dra. D^a. Nieves Cubo Mateo, D. Pablo Yuste Rodríguez-Escalona

1. COMPETENCIAS Y RESULTADOS DE APRENDIZAJE

1.1. Competencias

Competencias básicas

- CB1, CB2, CB3, CB4, CB5.

Competencias generales

- CG2. (Aplicar) Saber aplicar los conocimientos adquiridos en la definición y planteamiento de problemas y en la búsqueda de sus soluciones en contextos matemáticos y no matemáticos.

Competencias transversales

- CT1. (Comunicar) Comunicar de forma oral o escrita información, ideas, problemas y soluciones a un público tanto especializado como no especializado.
- CT2. (Usar herramientas) Identificar y conocer las herramientas informáticas básicas que sirven como instrumento de apoyo al trabajo académico y profesional.
- CT5. (Trabajar en equipo) Saber trabajar en equipo contribuyendo de forma activa al resultado de problema o proyecto a resolver.

Competencias específicas:

- CE6. (Modelizar) Utilizar las herramientas matemáticas más adecuadas a los fines que se persigan para proponer, analizar, validar e interpretar modelos matemáticos sencillos.
- CE7. (Instrumentalizar) Utilizar aplicaciones informáticas adecuadas para experimentar en matemáticas, resolver problemas y manejar modelos matemáticos.

- CE8. (Programar) Desarrollar programas que resuelvan problemas o modelos matemáticos utilizando cada caso el entorno computacional adecuado.
- CE9. (Interpretar) Adquirir conocimiento básico de otras disciplinas para saber interpretar en términos matemáticos situaciones complejas provenientes de estas.

1.2. Resultados de aprendizaje

El estudiante al finalizar esta materia deberá:

- Adquirir conocimientos generales de programación y tipos de lenguajes.
- Diseñar algoritmos para resolver problemas científico-técnicos de complejidad variada.
- Conocer algoritmos básicos con datos elementales y estructurados.
- Utilizar un entorno de programación dinámica para codificar y ejecutar programas.
- Dadas unas especificaciones, diseñar e implementar un programa.

2. CONTENIDOS

2.1. Requisitos previos

Ninguno.

2.2. Descripción de los contenidos

Historia de la computación. Metodología de la programación. Tipos de datos y expresiones, conceptos de entrada y salida y estructura básica de un programa. Estructuras de control secuencial, condicional e iterativas. Introducción a listas, vectores, matrices, así como a sus operaciones básicas. Entorno para la edición, compilación, ejecución y depuración. Uso de un lenguaje dinámico para computación numérica y científica (por ejemplo, Python o Julia).

2.3. Contenido detallado

Presentación de la asignatura

Explicación de la **guía docente**

- Historia de la computación – Breve introducción.
- Metodología de la programación - General.
- Entorno para la edición, compilación, ejecución y depuración (Python).
- Tipos de datos y expresiones. Conceptos de entrada y salida. Estructura básica de un programa. Memoria estática y dinámica.
- Introducción a listas, tuplas, sets, diccionarios, vectores, matrices, y operadores.

- Introducción a numpy y matplotlib. Representación de datos.
- Estructuras de control secuencial, condicional y bucles.
- Funciones. Argumentos y datos de retorno.
- Aplicaciones de la programación en IA, ejemplos matemáticos (Aprendizaje supervisado [machine learning]: Regresión, función de costes, gradiente descendiente).

2.4. Actividades dirigidas

Durante el curso se realizarán varias actividades dirigidas en forma de trabajos orientados al aprendizaje y aplicación de la programación o ampliación de éstos.

La actividad formativa “Prácticas” será el marco para establecer contenido y desarrollo de estas actividades que los estudiantes completaran de forma individual o en grupo. La entrega y la asistencia a las actividades y/o prácticas es obligatoria. La falta de asistencia a una práctica conlleva automáticamente el suspenso de la asignatura en caso de que la ausencia no esté debidamente justificada.

2.5 Actividades formativas

CÓDIGO	ACTIVIDAD FORMATIVA	HORAS	PORCENTAJE DE PRESENCIALIDAD
AF1	Clases de teoría y problemas	45	100%
AF2	Tutorías	15	70%
AF3	Prácticas	12	100%
AF4	Estudio individual y trabajo autónomo	60	0%
AF5	Trabajos individuales o en grupo	12	0%
AF6	Evaluación	6	100%

3. SISTEMA DE EVALUACIÓN

3.1. Sistema de calificaciones

El sistema de calificaciones (R.D. 1125/2003, de 5 de septiembre) será el siguiente:

- 0 - 4,9 Suspenso (SS)
- 5,0 - 6,9 (Aprobado (AP)
- 7,0 - 8,9 Notable (NT)
- 9,0 - 10 Sobresaliente (SB)

La mención de «Matrícula de Honor» podrá ser otorgada a alumnos que hayan obtenido una calificación igual o superior a 9,0. Su número no podrá exceder del cinco por ciento de los alumnos matriculados en una materia en el correspondiente curso académico, salvo que el número de alumnos matriculados sea inferior a 20, en cuyo caso se podrá conceder una sola «Matrícula de Honor».

3.2. Criterios de evaluación

Convocatoria ordinaria

Sistemas de evaluación	Porcentaje
SE1 Prueba parcial	15%
SE2 Examen final	55%
SE3 Presentación de trabajos	30%

Convocatoria extraordinaria

Sistemas de evaluación	Porcentaje
SE2 Examen final	70%
SE3 Presentación de trabajos	30%

3.3. Restricciones

Calificación mínima

Las ponderaciones anteriores sólo se aplicarán si el alumno/a obtiene al menos un 4 en el examen final.

Es imprescindible la entrega de todos los trabajos y prácticas propuestas en la asignatura. Para poder hacer media de los trabajos/prácticas es necesario obtener en cada uno de ellos una nota igual o superior a 3.5 puntos, y la nota media de todos los trabajos/prácticas deber ser superior o igual a 5. La no superación de los trabajos/prácticas supone el suspenso automático de la asignatura.

La convocatoria extraordinaria consiste en un examen sobre los contenidos de la asignatura desarrollados en las clases de teoría y problemas. Este examen pondera un 70%, el resto de la nota final corresponde a la calificación de las entregas de trabajos evaluables solicitados durante el periodo docente. Si estos trabajos están suspensos en la convocatoria ordinaria, pueden ser recuperados en convocatoria extraordinaria previa petición del estudiante al profesor. Esta petición se debe realizar por escrito en un plazo máximo de 10 días después de la publicación de la nota final de la convocatoria ordinaria. Esta ponderación también se aplica solo en el caso de que el alumno obtenga al menos un 4 en este examen final.

Asistencia

El alumno que, injustificadamente, deje de asistir a más de un 25% de las clases presenciales podrá verse privado del derecho a examinarse en la convocatoria ordinaria.

Normas de escritura

Se prestará especial atención en los trabajos, prácticas y proyectos escritos, así como en los exámenes tanto a la presentación como al contenido, cuidando los aspectos gramaticales y ortográficos. El no cumplimiento de los mínimos aceptables puede ocasionar que se resten puntos en dicho trabajo.

3.4. Advertencia sobre plagio

La Universidad Antonio de Nebrija no tolerará en ningún caso el plagio o copia. Se considerará plagio la reproducción de párrafos a partir de textos de auditoría distinta a la del estudiante (Internet, libros, artículos, trabajos de compañeros...), cuando no se cite la fuente original de la que provienen. El uso de las citas no puede ser indiscriminado. El plagio es un delito.

En caso de detectarse este tipo de prácticas, se considerará Falta Grave y se podrá aplicar la sanción prevista en el Reglamento del Alumno.

4. BIBLIOGRAFÍA

Bibliografía básica

- <https://www.sololearn.com/>
- *A Byte of Python*. Free online book: <https://python.swaroopch.com/>

Bibliografía para prácticas

- <https://personal.math.ubc.ca/~pwalls/math-python/>

Bibliografía complementaria

- <https://www.oreilly.com/library/view/machine-learning-with/9781491989371/ch01.html>
- Sam Morley. *Applying Math with Python: Practical Recipes for Solving Computational Math Problems Using Python Programming and Its Libraries*. Packt Publishing (2020).