

DATOS DE IDENTIFICACIÓN

Titulación:	Grado en Análisis de Negocios/Business Analytics
-------------	--

Rama de Conocimiento:	Ciencias Sociales y Jurídicas
-----------------------	-------------------------------

Facultad/Escuela:	Derecho, Empresa y Gobierno
-------------------	-----------------------------

Asignatura:	Algoritmos
-------------	------------

Tipo:	Obligatoria
-------	-------------

Créditos ECTS:	3
----------------	---

Curso:	2
--------	---

Código:	5325
---------	------

Periodo docente:	Tercer semestre
------------------	-----------------

Materia:	Informática Aplicada al Análisis de Negocios / IT applied to Business Analytics
----------	---

Módulo:	Formación Disciplinar
---------	-----------------------

Tipo de enseñanza:	Presencial
--------------------	------------

Idioma:	Castellano
---------	------------

Total de horas de dedicación del alumno:	75
--	----

Equipo Docente	Correo Electrónico
Raúl Antonio del Águila Escobar	raul.delaguila@ufv.es

DESCRIPCIÓN DE LA ASIGNATURA

La asignatura Algoritmos introduce al alumno en la resolución de problemas gracias a la aplicación del pensamiento estructurado a través de conceptos de programación básicos, haciendo especial hincapié en la presentación de las técnicas algorítmicas esenciales y la utilización de diferentes paradigmas, lenguajes y entornos de programación para la construcción de aplicaciones informáticas, aplicando metodologías de manera rigurosa que proporcionen al alumno buenos hábitos en el desarrollo de software.

Se imparte en el primer semestre del segundo curso de los estudios de Grado en Business Analytics, y requiere

de una dedicación de 70 horas por parte del alumno.

La asignatura se centra en la resolución de tareas, construyendo programas que resuelvan problemas sencillos. Para ello parte del diseño del algoritmo, utilizando estructuras modulares y distintos métodos de programación, y alcanza tanto la implementación del mismo en un lenguaje de programación estructurado, utilizando las estructuras de control fundamentales de la codificación y tipos de datos básicos, como su posterior compilación, depuración y documentación. El lenguaje que se utilizará para transmitir los contenidos es Python.

La asignatura comienza con una breve introducción a los conceptos básicos relacionados con la arquitectura de ordenadores y sistemas operativos, para luego abordar la resolución sistemática de tareas mediante el diseño de algoritmos que solucionen problemas sencillos. Para ello, el alumno parte de un conocimiento teórico sobre la estructuras de control y modulares presentes en cualquier programa informático, que expresará en el diseño de un algoritmo en pseudocódigo que, posteriormente, puede ser traducido de forma directa a Python 3.

OBJETIVO

El objetivo final de la asignatura consiste en presentar a los estudiantes los principios de la resolución sistemática de problemas a través del diseño de algoritmos y la codificación de una solución, siguiendo las reglas básicas de la computación.

El alumno expresará la solución en pseudocódigo. Esta formalización inicial del algoritmo seguirá las reglas sintácticas de Python 3, de modo que posteriormente ese algoritmo puede ser expresado de forma directa en este lenguaje de programación. De este modo, el alumno se empieza a familiarizar con un modo de expresión algorítmica similar a un lenguaje moderno y a priori sencillo.

El producto resultante, programa o algoritmo, es fruto del carácter de artista que posee el desarrollador, en este caso el alumno. Por ende, esta asignatura plantea acompañar al alumno en el descubrimiento de que su sed científica es su humanidad, produciéndose un encuentro de búsqueda entre formador y alumno, abriendo las puertas para poder afrontar así las preguntas antropológicas propias del hombre como científico e ingeniero informático, en torno al qué, el para qué y las consecuencias de los actos humanos, sociales, políticos y económicos.

CONOCIMIENTOS PREVIOS

Se recomienda disponer de los conocimientos básicos de lógica, álgebra y aritmética.

CONTENIDOS

Tema 1. Introducción a la arquitectura de computadores.

- Introducción a la arquitectura de un computador básico
- Introducción a los Niveles de computación.
- Paradigmas de programación.

Tema 2. Algoritmos.

- Introducción al concepto de algoritmo.
- Algoritmo vs programa informático.
- Elementos presentes en un algoritmo.
- Variables, identificadores, expresiones.
- Tipos de datos y clases de tipado.
- Estructuras de control.



- Funciones.
- Recursividad.

Tema 3. Introducción al concepto de Tipo Abstracto de Datos.

- Concepto.
- Arrays y matrices.

Tema 4. Algoritmos de ordenación y búsqueda.

- Introducción.
- Principales algoritmos.

ACTIVIDADES FORMATIVAS

La metodología seguida en esta asignatura está enfocada en un aprendizaje basado en problemas, donde para alcanzar los resultados de aprendizaje propuestos se combinan lecciones expositivas, clases prácticas, clases invertidas y sesiones de trabajo colaborativo. El objetivo principal de esta metodología es llevar a cabo un aprendizaje basado en la resolución de pequeños ejercicios para poner en práctica los conceptos asimilados. Durante las clases, tras una breve introducción teórica, el profesor planteará un problema que deberá ser solucionado por el alumno.

Existen dos tipologías fundamentales de problemas en esta asignatura: problemas orientados al entendimiento de las estructuras de control y de datos, y diseño de algoritmos. Estos problemas pueden ser solucionados de tres formas diferentes: individualmente, en parejas o colectivamente mediante la guía del profesor, quien establecerá los criterios de validez de una solución sobre la base de una verificación de la solución. Durante el proceso de solución del problema, los alumnos deberán estructurar la solución de acuerdo a las especificaciones definidas en el planteamiento que el profesor realizará, sin perjuicio de las mejoras metodológicas, prácticas o de técnica algorítmica introducidas por el alumno.

En resumen, durante las clases prácticas se fomenta la discusión y el trabajo en equipo con apoyo del profesor para facilitar el aprendizaje, donde exista un ambiente de trabajo óptimo para complementar y culminar el trabajo autónomo del alumno.

Dado que esta metodología requiere un alto grado de organización, se utilizará el Aula Virtual como plataforma de comunicación, soporte de materiales de estudio y como repositorio de prácticas y ejercicios.

DISTRIBUCIÓN DE LOS TIEMPOS DE TRABAJO

ACTIVIDAD PRESENCIAL	TRABAJO AUTÓNOMO/ACTIVIDAD NO PRESENCIAL
30 horas	45 horas
Lección Expositiva 6h Laboratorio 16h Presentación y Defensa de Trabajos 2h Tutorías 2h Evaluación 4h	Estudio y Trabajo Individual 30h Trabajo en Grupo 15h

COMPETENCIAS

Competencias básicas



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

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

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

Que los estudiantes puedan transmitir información, ideas, problemas y soluciones a un público tanto especializado como no especializado

Que los estudiantes hayan desarrollado aquellas habilidades de aprendizaje necesarias para emprender estudios posteriores con un alto grado de autonomía

Competencias generales

Capacidad de organización, sistematización y planificación en la identificación de problemas, pautas y modelos en el contexto del big data

Capacidad para el cumplimiento de objetivos, resolución de problemas y toma de decisiones en un entorno de datos masivos tanto cuantitativos como cualitativos

Capacidad de diseñar e implementar proyectos e informes, utilizando con naturalidad los canales digitales

Competencias específicas

Conocer, comprender y aplicar los fundamentos y las herramientas de la programación para el uso y explotación de la información, garantizando su seguridad e integridad

RESULTADOS DE APRENDIZAJE

Analiza e interpreta el problema a resolver y elabora algoritmos utilizando las estructuras adecuadas de control y de datos, siguiendo el paradigma de programación estructurada, utilizando tipos de datos elementales y estructuras de control básicas.

Demuestra destreza en el uso de las herramientas necesarias para el correcto desarrollo del software, incluyendo librerías, control de versiones, documentación, etc. Depura los errores detectados y analiza las posibles alternativas de resolución del problema.



Emplea las estructuras de iteración adecuadas y maneja correctamente la entrada y salida de información con el sistema de archivos.

Identifica fallos en la lógica de un algoritmo, presentando soluciones alternativas a la resolución del problema.

SISTEMA DE EVALUACIÓN DEL APRENDIZAJE

El sistema de evaluación contempla varios tipos de pruebas, distribuidos de la siguiente manera:

Convocatoria Ordinaria:

- Examen final de tipo teórico-práctico: 70 %.
- Realización de trabajos y prácticas: 25 %.
- Participación e implicación en la asignatura: 5 %.

Convocatoria Extraordinaria:

- Examen final de tipo teórico-práctico: 70 %.
- Realización de trabajos y prácticas: 25 %.
- Participación e implicación en la asignatura: 5 %.

De cara a la convocatoria extraordinaria, se guardarán las notas de:

- Examen final de tipo teórico-práctico, siempre y cuando se haya obtenido un 5 o más en la convocatoria ordinaria.
- Realización de trabajos y prácticas, siempre y cuando se haya obtenido un 5 o más en la convocatoria ordinaria.
- Participación e implicación en la asignatura.

Alumnos Dispensados: serán evaluados por el mismo tipo de pruebas. El 5 % correspondiente a la participación e implicación se evaluará en base a la entrega de los ejercicios correspondientes en la fecha establecida.

Criterios para superar la asignatura:

- Obtener una nota igual o superior a 5 puntos sobre 10 en el Examen final de tipo teórico-práctico.
- La entrega de prácticas y trabajos no es obligatoria, pero tendrán una calificación de 0 de cara al cálculo de la media correspondiente. Las prácticas tendrán una fecha de entrega. Si se entregan fuera de fecha, tendrán una penalización de un 30%, siempre y cuando se entreguen como tarde, una semana después de la fecha establecida.
- La asignatura se considerará aprobada en la convocatoria correspondiente si la nota final es igual o superior a 5 puntos sobre 10.

Notas adicionales:

- La entrega del examen por parte del profesor el día del examen implica automáticamente la presentación a la convocatoria correspondiente.
- A efecto de cómputo de convocatorias en una asignatura, solamente se contabilizarán como consumidas aquellas en las que el alumno se haya presentado a todas las pruebas de evaluación, o a una parte de las mismas, siempre que su peso en la nota final supere el 50 %, aunque no se presente al examen final. Se entenderá que un alumno se ha presentado a una prueba aunque la abandone una vez comenzada la misma. La condición de No Presentado en la convocatoria extraordinaria estará ligada a la no asistencia o entrega de ninguna prueba, práctica o trabajo que esté pendiente.
- Las conductas de plagio, así como el uso de medios ilegítimos en las pruebas de evaluación, serán sancionados conforme a los establecido en la Normativa de Evaluación y la Normativa de Convivencia de la universidad.

BIBLIOGRAFÍA Y OTROS RECURSOS

Básica

Luis Joyanes Aguilar. Fundamentos de programación: algoritmos, estructuras de datos y objetos / 5ª ed. México :McGraw-Hill Interamericana,2020.

Complementaria

Kent D. Lee · Steve Hubbard Data Structures and Algorithms with Python 2nd, Springer

