

TÉCNICO SUPERIOR UNIVERSITARIO EN TECNOLOGÍAS COMPUTACIONALES
Modalidad Escolarizada

Programa de Estudios

Lenguajes de Programación

NOMBRE DE LA ASIGNATURA Lenguajes de Programación

CLAVE DE LA ASIGNATURA TTC702

CICLO SÉPTIMO TETRAESTRE

CRÉDITOS	HORAS FRENTE A DOCENTE	HORAS INDEPENDIENTES
5	48	32

OBJETIVO(S) GENERAL(ES) DE LA ASIGNATURA

Al terminar el curso el alumno entenderá las diferencias y las similitudes entre diferentes lenguajes y paradigmas de programación. Podrá también tener elementos teóricos para elegir entre lenguajes y paradigmas de acuerdo al problema que enfrente.

COMPETENCIAS:

- Diseña algoritmos computacionales que resuelven problemas específicos.
- Diseña sistemas informáticos que satisfacen las necesidades de alguna organización.

INTRODUCCIÓN A LA ASIGNATURA:

Los lenguajes de programación de alto nivel nos permiten describir algoritmos con un lenguaje más cercano al lenguaje humano, o de las matemáticas, con lo que podemos escribir programas más sofisticados que usando únicamente el lenguaje de máquina, que para nosotros es muy difícil de entender.

Algunos de estos lenguajes modernos satisfacen las demandas de los llamados “Paradigmas de Programación”. Por ejemplo, Java es un lenguaje orientado a objetos y C es un lenguaje estructurado.

En esta asignatura se verán los principios de algunos paradigmas de programación diferentes a los ya cubiertos en los cursos de Programación y Estructuras de Datos, que son el paradigma estructurado y el paradigma orientado a objetos. Estos paradigmas son utilizados de manera menos frecuente, pero nos dan una forma diferente de organizar un programa. Algunas veces estos paradigmas nos permiten resolver cierto tipo de problemas de manera más eficaz.

Primero cubriremos algunos aspectos generales de los lenguajes de programación y después veremos dos paradigmas diferentes.

En la primera parte veremos qué es lo que los lenguajes de programación tienen en común. El primer paradigma que veremos es el de la programación funcional, que utiliza el concepto de función como “ciudadano de primera”, de tal manera que las funciones se pueden pasar como parámetros y devolver como el resultado de otra función.

Unidades temáticas

Nombre de la unidad	
I.	Conceptos básicos.
II.	Programación Funcional
III.	Programación Lógica

Técnica didáctica

Aprendizaje Basado en problemas	Aprendizaje basado en proyectos	Aprendizaje Colaborativo	Aprendizaje experimental	Otra
		X		

Estrategia de evaluación

Componente	Porcentaje
Exámenes parciales (2)	50
Examen final	30
Tareas y actividades	20
Proyectos	
Asistencia	
Taller	
Total	100

Bibliografía

- PROGRAMMING LANGUAGES. PRINCIPLES AND PRACTICE. Kenneth C. Loudon. Course Technology. 2006
- Programming in Haskell. Graham Hutton. Cambridge University Press. 2007.
- A Gentle Introduction to Haskell Version 98. <https://www.haskell.org/tutorial/>.
- ¡Aprende Haskell por el bien de todos! <http://aprendehaskell.es/main.html>.
- L. Sterling y E. Shapiro. The Art of Prolog. MIT Press, second edition, 1994.
- T. Van Le. Techniques of Prolog Programming (with implementation of logical negation and quantified goals). John Wiley, 1993.
- José A. Alonso Jiménez. Introducción a la programación lógica con Prolog. 2006. <http://www.cs.us.es/~jalonso/>. De este texto se obtuvo una parte de las notas sobre Prolog.