



## **PROGRAMA ANALÍTICO**

### **LINEAMIENTOS GENERALES**

La presente asignatura es una actividad curricular que pertenece al décimo semestre de la carrera de Ingeniería en Computación, siendo una materia selectiva.

Existen un sinnúmero de lenguajes de programación disponibles en la industria, cada uno de ellos destinados a brindar facilidades para la resolución de ciertos tipos de problemas y diseñados con distintas características, todos basados en fundamentos teóricos que son propios de diferentes paradigmas. Básicamente, se trata de enseñar al alumno que no solo existe el modelo de programación imperativo tradicional, sino que también hay otros paradigmas, cuya respectivas implementaciones a su vez dan origen a lenguajes de programación con características propias.

La materia tiene entonces un agudo perfil teórico orientado a estudiar las características que definen a los lenguajes de programación: la estructuración de los datos, la estructuración de los comandos y los modelos de ejecución. Pero además debe estudiarse implementaciones en diversos lenguajes de los conceptos teóricos, por lo que una ejercitación de programación en varios lenguajes es necesaria, de forma tal que se analice para cada lenguaje en particular, el reflejo en su diseño de los modelos teóricos y cuales son las facilidades que brinda para ciertos tipos de problema.

Todo ello redundando en que el alumno adquiera capacidades para poder enfrentarse a nuevos lenguajes de programación con fundamentos teóricos que le faciliten su aprendizaje, y a que disponga de conocimientos que le permitan seleccionar el uso del lenguaje más apropiado a la problemática que se enfrente.

Es por estas razones que los ejes principales de la materia son el conocimiento teórico de los paradigmas de programación imperativo, funcional y la orientación a objetos, las diversas formas en que pueden estructurarse los datos y el código, por un lado, y por otro el estudio y ejercitación en programación de lenguajes que tengan una concepción basada en dichos paradigmas.

Los contenidos teóricos a impartir corresponden a estos conceptos:

- Modelos de ejecución y compilación de los programas.
- Sintaxis y semántica de los lenguajes. Forma BNF. Expresiones Regulares.
- Binding: atributos que caracterizan a las variables y al código
- Estructuración de los Datos: desarrollando temas de:
  - Tipos de variables, visibilidad y alcance
  - Sistema de tipos. Lenguajes fuertemente o débilmente tipados.
  - Compatibilidad de tipos.
  - Clases, objetos y herencia
- Estructuración del código:

- Estructuras de Control
- Excepciones
- Rutinas: atributos y alcance. Formas de agruparlas.
- Genericidad de las rutinas,
- Paralelismo. Threads. Estructuras de control de la concurrencia.
- Paradigma funcional:
  - Valores, funciones. Cálculo Lambda. Ligaduras dinámicas. Tipado dinámico.
  - Cálculo lambda. Funciones no estrictas. Mónadas
- Orientación a Objetos:
  - Clases, objetos y tipos de herencia.
  - Encapsulamiento de los datos.
  - Polimorfismo y genericidad de métodos.

## **METODOLOGIA DE ENSEÑANZA**

Las clases impartidas son teóricas, y prácticas de laboratorio.

Las actividades teóricas se realizan a través de exposiciones de los docentes orientadas a desarrollar en los alumnos conocimientos sobre las estructuras genéricas de datos y la casuística donde son útiles y eficientes.

Durante el desarrollo de las clases prácticas de laboratorio se realizan actividades de programación en lenguajes que implementen los paradigmas teóricos transmitidos, que le permiten al estudiante poner en práctica los conocimientos adquiridos y ejercitarse en la selección de alternativas acorde a los requerimientos de los problemas a solucionar.

## **EVALUACIÓN**

### **Condiciones para la promoción de la materia**

1. Tener aprobadas las materias correlativas.-
2. Asistir al 80% de las clases teóricas y prácticas.-
3. Rendir y aprobar un examen teórico con 50%, con posibilidad de recuperación.
4. Aprobar dos entre tres proyectos de programación en lenguajes diferentes.
5. Aprobar el trabajo final integrador.

Los alumnos que cumplan las exigencias referidas en los puntos 1 al 4 serán considerados regulares. Si además cumplen con el punto 4, se los consideraran como promocionados. El resto será considerado libre.

## CONTENIDOS TEMATICOS

### **Unidad 1: Lenguajes de Programación.**

Modelos de ejecución y compilación de los programas. Lenguajes interpretados y Lenguajes ejecutables. Máquina Virtual. Sintáxis de los lenguajes. Forma BNF. Expresiones Regulares. Semántica: significado de una expresión. Equivalencia de expresiones. Tipos de semántica. Ejemplos.

### **Unidad 2: Elementos Constitutivos de un Lenguaje. Variables.**

Concepto de Binding. Variables: sus atributos. Alcance y vida. Tipos. Tipos de variables, visibilidad y alcance. Sistema de tipos. Lenguajes fuertemente o débilmente tipados. Compatibilidad de tipos. Ejemplos.

### **Unidad 3: Elementos Constitutivos de un Lenguaje. Rutinas.**

Rutinas. Atributos de las rutinas. Tipos de paso de parámetros. Alcance. Formas de agruparse. Genericidad de las rutinas. Paralelismo. Threads. Estructuras de control de la concurrencia. Estructuras de control. Excepciones.

### **Unidad 4: Orientación a Objetos.**

Clases, objetos y tipos de herencia. Diferencia entre sistema de tipos y sistema de clases. Encapsulamiento de los datos. Polimorfismo y genericidad de métodos. Introducción a patrones de Diseño. Estudio de un lenguaje orientado a objetos y ejercitación.

### **Unidad 5: Lenguajes Funcionales.**

Valores, funciones. Cálculo Lambda. Ligaduras dinámicas. Tipado dinámico. Cálculo lambda. Funciones no estrictas. Mónadas. Estudio de un lenguaje funcional y ejercitación.

### **Unidad 5: Otro lenguaje imperativo.**

Estudio y ejercitación en un lenguaje imperativo pero con características particulares, como puede ser débilmente tipado, *scripting languages*, con tipado dinámico.

**DISTRIBUCION DE LA CARGA HORARIA**

<b>ACTIVIDAD</b>	<b>HORAS</b>
TEÓRICA	36
FORMACIÓN PRACTICA: CLASES PRACTICAS LABORATORIO	
	36
<b>TOTAL DE LA CARGA HORARIA</b>	<b>72</b>

**DEDICADAS POR EL ALUMNO FUERA DE CLASE**

<b>ACTIVIDAD</b>	<b>HORAS</b>
PREPARACION TEÓRICA	30
PREPARACION PRACTICA	60
<b>TOTAL DE LA CARGA HORARIA</b>	<b>90</b>

## **BIBLIOGRAFIA**

**1.) “Programming Languages Concepts”**

Autores: Ghezzi y Jazayeri. Editor: J. Wiley (1998)

**2.) “Lenguajes de Programación – Principios y Práctica”**

Autor: K. Louden. Editor: Thomson (2004)

**3.) “Programming Language Design Concepts”**

Autor: D. Watt. Editor: Wiley (2004)

**4.) “Concepts, Techniques and Models of Computer Programming”**

Autores: P. Van Roy, S. Haridi. Editor: MIT Press (2004)

**5.) “Concepts in Programming Languages”**

Autor: J. Mitchell. Editor: Cambridge U. Press (2003)

**6.) “Haskell: The craft of functional programming”**

Autor: S. Thomson Editor: Addison Wesley (1996)

**7.) “Programming Perl ”**

Autor: L. Wall . Editor: O'Reilly (1994)

**8.) “Smalltalk by Example”**

Autor: Alec Sharp. Edición electrónica (2004)