



**UNIVERSIDAD SIMÓN BOLÍVAR**  
**Vicerrectorado Académico**

1. Departamento: *COMPUTACIÓN Y TECNOLOGÍA DE LA INFORMACIÓN*

**2. Asignatura: LENGUAJES DE PROGRAMACION**

3. Código de la asignatura: CI-3641

No. de unidades-crédito: 3

No. de horas semanales: Teoría 3 Práctica Laboratorio

4. Fecha de entrada en vigencia de este programa: SEPTIEMBRE 89

**5.- REQUISITOS**

Algoritmos y Estructuras de Datos II (CI-2612)

Sistemas de Programas (CI-3711)

**6.- OBJETIVOS**

**6.1. Generales:**

Estudio de las características fundamentales de los lenguajes de alto nivel y de los paradigmas de programación que éstos implementan.

**6.2 Específicos:**

Al término del curso se espera que los estudiantes del mismo hayan alcanzado los siguientes objetivos:

- Estudio de los lenguajes de programación y de los principales paradigmas de computación.
- Familiarización con la nociones básicas del modelo imperativo de programación.
- Estudio del paradigma de programación con objetos y de las alternativas ofrecidas por los lenguajes de alto nivel para la implementación de tipos abstractos.

- Familiarización con los conceptos y técnicas básicas de la programación concurrente.
- Conocimiento de las alternativas propuestas por los paradigmas funcional y lógico.
- Capacidad de comparación entre distintos lenguajes y su adecuación para la resolución de problemas específicos.
- Establecimiento de parámetros de comparación de lenguajes de programación.

## **7. CONTENIDO.**

### 1. Introducción

Introducción histórica. sintaxis y semántica. Lenguajes de alto, medio y bajo nivel. Lenguajes de propósito general y de propósito específico. Paradigmas de programación.

### 2. Programación Imperativa.

Arquitectura de Von Neumann. Variables y estados de memoria. Flujo de control.

### 3. Variables y estados de memoria.

Tipo, valor, alcance y tiempo de vida de una variable. Variables locales y globales. Alcance y contexto: Modelo estático y dinámico. Tipos de pasaje de parámetros .

Tipos de datos abstractos y programación con objetos. Evolución del concepto de tipo. Encapsulamiento y privacidad. Objetos y clases. Generalidad, sobrecarga de operadores y polimorfismo. Reusabilidad. Modelo dinámico de pasaje de mensajes.

### 4. Flujo de Control.

Transferencias generales. Jerarquías de estructuras de control. Transformaciones entre modelos. programación estructurada. Transferencias entre unidades: subrutinas, corrutinas, unidades concurrentes.

Programación concurrente. Procesos e interacción entre procesos. Concurrencia y no determinismo. Paralelismo explícito e implícito. Mecanismo de sincronización y exclusión mutua. “Deadlocks”. Sincronización entre procesos que comparten memoria. Procesos comunicantes.

#### 5. Programación Lógica.

Lenguajes declarativos o no procedimentales. Lenguajes lógicos y demostración de teoremas. Bases de conocimiento. Mecanismo de inferencia.

#### 6. Programación Funcional

Características esenciales. Lenguajes aplicativos: objetos, funciones y formas funcionales. Elementos funcionales de lenguajes tradicionales. Lenguajes parcialmente funcionales.

#### 7. Lenguajes Comparados

Comparación entre lenguajes. Paralelismo implícito y explícito en modelos no imperativos. Lenguajes multiparadigmáticos. Elementos de análisis y diseño de lenguajes de programación.

### **8. BIBLIOGRAFIA**

Andrews y Schneider 83

Andrews, G.R. y Schneider, F.B.: “Conceptos and Notations for Concurrent programming”, Computing Surveys, Vol. 15, No. 1, Marzo 1983, pp. 3-40.

Ben - Ari 82

Ben-Ari, M.: Principles of Concurrent Programming, Segunda Edición John & Sons. 1987.

Ghzzi y Jazayeri 87

Ghezzi, C. y Jazayeri, M.: Programming Language Concepts, Segunda Edición, John Wiley & Sons. 1987.

Ledgard y Marcotty 75

ledgard, H.F. y Marcotty, M.: “A Generalogy of Control Structures”, CACM, Vol. 18, No. 11, Noviembre 1975, pp. 629-639.