

**UNIVERSIDAD AUTÓNOMA DE BAJA CALIFORNIA  
 COORDINACIÓN DE FORMACIÓN BÁSICA  
 COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN  
 UNIVERSITARIA  
 DEPARTAMENTO DE ACTUALIZACIÓN CURRICULAR Y FORMACIÓN  
 DOCENTE**

**DESCRIPCIÓN GENÉRICA DE UNIDADES DE APRENDIZAJE**

Descripción Genérica

Nombre: **Diseño de algoritmos**  
 Etapa: **Básica**  
 Área de conocimiento: **Programación e ingeniería de software**  
**Plan : 2008-1**

**Competencia:**

Diseñar algoritmos sencillos, legibles y comprensibles, mediante el uso de la heurística y técnicas de programación estructurada, para el entendimiento de las estructuras algorítmicas existentes y su comportamiento, con una actitud crítica y propositiva.

**Evidencia de desempeño:**

Se evaluara mediante exámenes teóricos, además se complementará la evaluación con practicas de laboratorio y resolución de problemas.

Presentar un trabajo final donde se desarrolle el análisis de una problemática, se diseñe una solución algorítmica, y se implemente un programa utilizando todas las técnicas y herramientas de programación vistas en el curso.

Distribución	HC	HL	HT	HPC	HCL	HE	CR	Requisito
	2	2	2			2	8	ninguno

Contenidos Temáticos

**1. Algoritmos y Programación**

- 1.1.- Conceptos básicos de programación
  - 1.1.1.- Diferencia entre programar y codificar.
- 1.2.- Análisis de problemas de lógica
- 1.3.- La trascendencia del modelo de Von Neuman

## **2. Representación de los algoritmos y el mundo de Karel**

- 2.1.- Especificación y estructura de un algoritmo
- 2.2.- El mundo del Robot Karel
  - 2.2.1. Instrucciones primitivas y programas simples.
  - 2.2.2. Crear instrucciones en Karel.
  - 2.2.3. Ejecución condicional de instrucciones.
  - 2.2.4. Ciclos en Karel.
  - 2.2.5. Programación avanzada del robot.
- 2.3. Compilador de Karel: Simulación y ejecución de algoritmos
- 2.4.- Uso de diagramas de flujo
  - 2.4.1. Elementos de un diagrama de flujo
  - 2.4.2. Representación de condicionales y ciclos
  - 2.4.3. Conectores
  - 2.4.4. Representación del diagrama de flujo con base en su algoritmo.

## **3. Estructuras de control y modularidad (divide y vencerás)**

- 3.1.- Constantes, variables y operadores matemáticos booleanos
- 3.2.- Operadores lógicos y operadores relacionales
- 3.3.- Secuencia, selección o condicionales, ciclos o iteraciones
- 3.4.- Subprogramas y/o procedimientos
- 3.5.- Estructura de algoritmos en pseudocódigo
- 3.6.- Introducción a las estructuras de datos
  - 3.6.1. Manejo de arreglos unidimensionales
  - 3.6.2. Manejo de arreglos multidimensionales
  - 3.6.3. Manejo de registros

## **4. Herramientas de diseño y aplicación de los algoritmos**

- 4.1. Programación avanzada mediante el uso de hojas de cálculo.
  - 4.1.1. Introducción a la herramienta
  - 4.1.2. Manipulación de datos numéricos y alfanuméricos.
  - 4.1.3. Manejo de tablas (filas y columnas)
  - 4.1.4. Fórmulas y funciones.
  - 4.1.5. Manejo de gráficos
  - 4.1.6. Solución de problemas
- 4.2. Programación avanzada mediante el uso de herramientas de cálculo numérico.
  - 4.2.1. Introducción a la herramienta.
  - 4.2.2. Comandos básicos de programación.
  - 4.2.3. Vectores y matrices.
  - 4.2.4. Manejo de gráficos.
  - 4.2.5. Manejo de funciones estándar
  - 4.2.6. Crear subprogramas
  - 4.2.7. Solución de problemas

## Referencias bibliográficas actualizadas

### **Básica**

Fundamentos de Programación  
(Algoritmos y Estructuras de Datos y Objetos)  
Luis Joyanes Aguilar  
McGraw Hill, 2003

Metodología de la Programación  
(Algoritmos, Diagramas de Flujo y Programas)  
Osvaldo Cairó  
Alfaomega,

Introducción Gradual a la Programación  
(El robot Karel)  
Richard E. Pattis  
Limusa, Noriega.

### **Complementaria**

Computación y Programación Moderna  
(Perspectiva integral de la informática)  
Guillermo Levine  
Addison Wesley

Programming in MATLAB  
Herniter, Marc E.  
Brooks/Cole-Thomson Learning,

Excel 2003 formulas  
Walkenbach, John.  
Wiley