

**UNIVERSIDAD AUTÓNOMA DE CHIAPAS  
LICENCIATURA EN SISTEMAS COMPUTACIONALES**

**Área de formación:** Disciplinaria

**Unidad académica:** Algoritmos

**Ubicación:** Primer semestre

**Clave:** \_\_\_\_\_

**Horas semana-mes:** 5

**Horas Teoría:** 3

**Horas Práctica:** 2

**Créditos:** 8

### **PRESENTACIÓN**

Los algoritmos son parte fundamental en la construcción del software, no se puede concebir la programación estructurada, orientada a objetos u otro tipo de programación sin las estructuras básicas: secuenciales, condicionales y repetitivas.

La algoritmia le brindará al estudiante las bases fundamentales para el curso de programación estructurada, así como para otras asignaturas ligadas con la ingeniería de software.

Asimismo, a través de estos conocimientos desarrollará la capacidad de abstracción, análisis y síntesis para el desarrollo de programas computacionales.

### **OBJETIVO GENERAL**

El alumno aplicará las técnicas de diseño de algoritmos computacionales para la construcción de programas en forma correcta y metodológica en la solución de problemas.

### **UNIDAD I.- CONCEPTOS BÁSICOS Y METODOLOGÍA PARA LA SOLUCIÓN DE PROBLEMAS COMPUTACIONALES**

**TIEMPO APROXIMADO:** 12 Horas

**OBJETIVO DE LA UNIDAD:** Aplicar los conceptos básicos de los algoritmos en la solución de problemas programables.

## **CONTENIDO**

- 1.1 Definición de algoritmo
- 1.2 Características de los algoritmos
- 1.3 Tipos de algoritmos
- 1.4 Procesos e instrucciones
  - 1.4.1 Procesos y su descripción
  - 1.4.2 Variables
- 1.5 Especificación de un algoritmo
  - 1.5.1 Predicados
  - 1.5.2 Elementos de una especificación
- 1.6 Metodología para la solución de problemas
  - 1.6.1 Definición del problema
  - 1.6.2 Análisis de los datos
  - 1.6.3 Diseño de la solución
  - 1.6.4 Codificación
  - 1.6.5 Prueba y depuración
  - 1.6.6 Documentación
  - 1.6.7 Mantenimiento

## **UNIDAD II.- ENTIDADES BÁSICAS PARA EL DESARROLLO DE ALGORITMOS**

**TIEMPO APROXIMADO:** 10 Horas

**OBJETIVO DE LA UNIDAD:** Construir los conceptos de entidades básicas para el desarrollo de algoritmos.

### **CONTENIDO**

- 2.1 Tipos de datos
- 2.2 Operadores y operandos
- 2.3 Expresiones
- 2.4 Identificadores

## **UNIDAD III.- TÉCNICAS DE PROGRAMACIÓN CON CALIDAD Y REPRESENTACIÓN DE ALGORITMOS**

**TIEMPO APROXIMADO:** 10 Horas

**OBJETIVO DE LA UNIDAD:** Aplicar técnicas de programación y representación de algoritmos con calidad.

## **CONTENIDO**

- 3.1 Técnicas de programación
  - 3.1.1 Top down
  - 3.1.2 Botton up
  - 3.1.3 Warnier
- 3.2 Representación de algoritmos
  - 3.2.1 Diagramas de flujo
  - 3.2.2 Pseudocódigo
- 3.3 Técnicas de programación con calidad
  - 3.3.1 Metodología
  - 3.3.2 Tamaño del producto
  - 3.3.3 Técnicas de estimación
  - 3.3.4 Agenda de trabajo y seguimiento a la agenda de trabajo
  - 3.3.5 Administración de los defectos

## **UNIDAD IV.- CONCEPTOS BÁSICOS DE PROGRAMACIÓN**

**TIEMPO APROXIMADO:** 10 Horas

**OBJETIVO DE LA UNIDAD:** Diseñar algoritmos y aplicar los conceptos esenciales para la elaboración de programas básicos.

## **CONTENIDO**

- 4.1. Componentes básicos de un lenguaje de programación
  - 4.1.1. Palabras reservadas
  - 4.1.2. Comentarios
  - 4.1.3. Tipos de datos
  - 4.1.4. Identificadores
  - 4.1.5. Operadores
- 4.2 Estructura general de un programa
- 4.3 Programas básicos

## **UNIDAD V.- ESTRUCTURAS DE CONTROL**

**TIEMPO APROXIMADO:** 15 Horas

**OBJETIVO DE LA UNIDAD:** Aplicar estructuras de control en el diseño de algoritmos básicos y convertirlos a un lenguaje de alto nivel.

## **CONTENIDO.**

- 5.1 Secuenciales
  - 5.1.1 Asignación
  - 5.1.2 Entrada
  - 5.1.3 Salida

- 5.2 Condicionales
  - 5.2.1 Simple
  - 5.3.2 Múltiple
- 5.3 Estructuras Repetitivas
  - 5.3.1 Mientras (while)
  - 5.3.5 Repetir (do while)
  - 5.3.3 Desde para (for)

## **UNIDAD VI.- PROCEDIMIENTOS Y FUNCIONES**

**TIEMPO APROXIMADO:** 23 Horas

**OBJETIVO DE LA UNIDAD:** Conocer las partes que integran los subprogramas.

### **CONTENIDO**

- 6.1 La importancia de las funciones
- 6.2 Funciones sin parámetros
- 6.3 Funciones que reciben y devuelven parámetros
- 6.4 Prototipos
- 6.5 Transferencia de parámetros desde el sistema operativo
- 6.6 Definición de subprogramas
- 6.7 Funcionamiento de subprogramas
- 6.8 Manipulación de subprogramas

### **EXPERIENCIAS DE APRENDIZAJE**

- Foro de discusión mediante el portal en Internet
- Elaboración de proyectos
- Ejercicios grupales fuera y dentro de clase
- Exposición de investigaciones y tareas por equipos de trabajo
- Planteamiento y solución de problemas
- Diseño de programas

### **CRITERIOS DE EVALUACIÓN**

Exámenes por unidad	50%
Investigación y tareas	30%
Proyecto Final	<u>20%</u>
Total	100%

## **BIBLIOGRAFÍA BÁSICA**

Tremblay, Jean Paul & Bunt, Richard B. (1998). Introducción a la ciencia de las computadoras. México: Mc. Graw Hill.

Joyanes, Luis (1998). Metodología de la programación. México: Mc. Graw Hill.

Castro Jorge y otros (1993). Curso de Programación. España: Mc Graw Hill.

Joyanes, Luis (1998). Fundamentos de programación. España: Mc. Graw Hill, 2da. Edición.

Weiss, Mark Allen (1999). Estructuras de datos y análisis de algoritmos. México: Addison\_Wesley Longman.

Joyanes, Luis (1998). Fundamentos de programación. España: Mc. Graw Hill.

Pratt, Terrence W. Y Marvin V. Zelkowitz (1997). Lenguajes de programación: diseño e implementación. México: Prentice Hall.

## **BIBLIOGRAFÍA COMPLEMENTARIA**

Humphrey, W. (1995). A Discipline for Software Engineering READING, MASS: Addison Wesley, SEI Series in Software Engineering.

Humphrey, W. (1997). Introduction to the Personal Software Process READING, MASS: Addison. Wesley, SEI Series in Software Engineering.