

### Historia del programa

Lugar y fecha de elaboración	Participantes	Observaciones (Cambios y justificaciones)
Cancún, Q. Roo, 21/04/2010	M.C. Joel Antonio Trejo Sánchez	Se cambió la estructura por actualización del temario.

### Relación con otras asignaturas

Anteriores	Posteriores
Asignatura(s) Algoritmos y Estructura de datos.	Asignatura(s) Programación Orientada a Objetos
Tema(s) a) Estructuras de datos estáticas. b) Estructuras de control.	Tema(s) a) Tipos de datos abstractos.

Nombre de la asignatura	Departamento o Licenciatura
Técnicas algorítmicas	Ingeniería en Telemática

Ciclo	Clave	Créditos	Área de formación curricular
1 - 1	IT0107	8	Profesional Asociado y Licenciatura Básica

Tipo de asignatura	Horas de estudio			
	HT	HP	TH	HI
Materia	32	32	64	64

### Objetivo(s) general(es) de la asignatura

### Objetivo cognitivo

Describir las principales estrategias de algoritmia y complejidad para el diseño de algoritmos.

### Objetivo procedimental

Usar técnicas de algoritmia para dar solución a problemas computables.

### Objetivo actitudinal

Fomentar la disciplina para el análisis y diseño de soluciones a problemas computables.

## Unidades y temas

---

### Unidad I. ESTRUCTURAS DE DATOS AVANZADAS

Describir las principales metodologías de algoritmia para el análisis de los principales algoritmos de ordenamiento y búsqueda.

1) Listas enlazadas

2) Tablas Hash

3) Árboles

4) Grafos

### Unidad II. INTRODUCCIÓN AL ANÁLISIS DE ALGORITMOS

Usar las principales metodologías de algoritmia para el análisis de los principales algoritmos de ordenamiento y búsqueda.

1) Análisis de complejidad

a) Tiempo de ejecución

b) Orden de crecimiento

c) Métodos de análisis de complejidad

2) Algoritmos de Ordenamiento

a) Insertion Sort

b) Bubble Sort

c) Quicksort

d) Heapsort

### 3) Algoritmos de Búsqueda

a) Búsqueda secuencial

b) Búsqueda binaria

## Unidad III. ALGORITMOS VORACES

Aplicar algoritmos voraces para la solución de problemas computables.

### 1) Definición

### 2) Algoritmos para grafos

a) Algoritmo de Kruskal

b) Algoritmo de Prim

c) Algoritmo de Dijkstra

## Unidad IV. DIVIDE Y VENCERÁS

Emplear la técnica de divide y vencerás para la solución de problemas computables.

### 1) Precedentes históricos

### 2) Recursividad y divide y vencerás

### 3) Algunos problemas utilizando divide y vencerás

## Unidad V. Programación Dinámica

Bosquejar la técnica de programación dinámica para la solución de problemas computables.

1) Definición

2) Aplicaciones

a) Problema de la mochila

b) Otros problemas

## Actividades que promueven el aprendizaje

### Docente

Promover el trabajo individual en la definición de propuestas de solución a problemas determinados.

Coordinar la discusión de casos prácticos.

Realizar demostraciones de algoritmos

Aplicar prácticas para la definición de algoritmos.

Definir estrategias para identificar las principales técnicas algorítmicas para resolver problemas computables.

### Estudiante

Realizar tareas asignadas

Participar en el trabajo individual y en equipo

Resolver casos prácticos

Discutir temas en el aula

Participar en actividades extraescolares

## Actividades de aprendizaje en Internet

El estudiante deberá acceder al portal (señalar las actividades que realizarán):

Se promoverá el uso de mecanismos asíncronos (correo electrónico, grupo de noticias, WWW y tecnologías de información) como medio de comunicación.

## Criterios y/o evidencias de evaluación y acreditación

### Criterios

Examen

### Porcentajes

30

Tareas	30
Evidencias individuales	20
Evidencias grupales	20
Total	100

## Fuentes de referencia básica

---

### Bibliográficas

Cormen. (1999) Introduction to Algorithms. (2da Edición). MIT Press

Donald E. Knuth. (2000) The Art of Computer Programming. (1era Edición). Addison Wesley

Gilles Brassard. (1999) Fundamentos de Algoritmos. (2da Edición) Prentice Hall

Steven Skiena. (2002) Programming Challenges. (2da Edición) Springer Verlag

Weiss. (2000) Data Structures and Problem Solving Using Java. (2da Edición) Addison Wesley

### Web gráficas

<http://domino.research.ibm.com/comm/research.nsf/pages/r.algorithms.html> 28 de Junio de 2010

## Fuentes de referencia complementaria

---

### Bibliográficas

Robert Sedgewick. (2005) Algorithms in C++ part 5. Graph Algorithms. (1era Edición) Addison Wesley

Vazirani. (2007) Approximation Algorithms. (2da Edición). Springer

### Web gráficas

No aplica

## Perfil profesiográfico del docente

---

### Académicos

Maestría en Ciencias de la computación, Maestría en Ingeniería en Sistemas

**Docentes**

Tener experiencia docente a nivel superior mínima de 3 años en ingeniería

**Profesionales**

Tener experiencia en desarrollo de sistemas