



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
FACULTAD DE ESTUDIOS SUPERIORES ACATLÁN

PLAN DE ESTUDIOS DE LA LICENCIATURA EN
MATEMÁTICAS APLICADAS Y COMPUTACIÓN



PROGRAMA DE ASIGNATURA

SEMESTRE:3 (TERCERO)

Estructuras de Datos

CLAVE:

MODALIDAD	CARÁCTER	TIPO	HORAS AL SEMESTRE	HORAS SEMANA	HORAS TEÓRICAS	HORAS PRÁCTICAS	CRÉDITOS
Curso	Obligatoria	Teórico-Práctica	96	6	4	2	10

ETAPA DE FORMACIÓN	Básico
CAMPO DE CONOCIMIENTO	Computación

SERIACIÓN	Indicativa
ASIGNATURA(S) ANTECEDENTE	Programación II
ASIGNATURA(S) SUBSECUENTE(S)	Bases de Datos

Objetivo general: El alumno aplicará las estructuras de datos, así como las técnicas de recuperación y ordenamiento de datos, en la implementación de algoritmos computacionales.

Unidad	Índice Temático	Horas	
		Teóricas	Prácticas
1	Introducción a los algoritmos de estructuras de datos	8	4
2	Almacenamiento estático, dinámico y estructuras elementales	16	8
3	Estructuras de datos lineales y no lineales	24	12
4	Técnicas de ordenamiento y búsqueda	16	8
Total de horas:		64	32
Suma total de horas:		96	

HORAS		UNIDAD	CONTENIDO
T	P		
8	4	1	<p>INTRODUCCIÓN A LOS ALGORITMOS DE ESTRUCTURAS DE DATOS</p> <p>Objetivo particular: El alumno reconocerá la importancia del manejo apropiado de las estructuras de datos durante la programación de algoritmos, así como las principales técnicas que existen para diseñarlos.</p> <p>Temas:</p> <ul style="list-style-type: none"> 1.1. Abstracción de datos 1.2. Clasificación de las estructuras de datos 1.3. Algoritmos y estructuras de datos 1.4. Técnicas de diseño de algoritmos 1.5. Recursividad
16	8	2	<p>ALMACENAMIENTO ESTÁTICO, DINÁMICO Y ESTRUCTURAS ELEMENTALES</p> <p>Objetivo particular: El alumno identificará las diferentes formas de almacenamiento de las estructuras de datos, sus ventajas y limitantes.</p> <p>Temas:</p> <ul style="list-style-type: none"> 2.1 Almacenamiento estático y dinámico 2.2 Arreglos <ul style="list-style-type: none"> 2.2.1 Arreglos de diferentes tipos de datos 2.2.2 Arreglos de punteros 2.2.3 Arreglos como parámetros en funciones 2.2.4 Arreglos unidimensionales, bidimensionales y multidimensionales 2.2.5 Arreglos dinámicos 2.2.6 Arreglos triangulares: inferior y superior 2.3 Registros <ul style="list-style-type: none"> 2.3.1 Arreglos de registros (estáticos y dinámicos) 2.4 Enumeraciones y colecciones
24	12	3	<p>ESTRUCTURAS DE DATOS LINEALES Y NO LINEALES</p> <p>Objetivo particular: El alumno aplicará las estructuras de datos lineales y no lineales en la implementación de algoritmos computacionales.</p> <p>Temas:</p> <ul style="list-style-type: none"> 3.1 Pilas <ul style="list-style-type: none"> 3.1.1 Definición y representación 3.1.2 Operaciones de inserción y extracción 3.1.3 Implementación dinámica con diferentes tipos de datos 3.1.4 Implementación con arreglos de diferentes tipos de datos 3.1.5 Notación infija, prefija y postfija. 3.2 Cola <ul style="list-style-type: none"> 3.2.1 Definición y representación 3.2.2 Operaciones de inserción y extracción 3.2.3 Implementación dinámica con diferentes tipos de datos

			<p>3.2.4 Implementación con arreglos de diferentes tipos de datos 3.2.5 Cola circular, doblemente ligada y de prioridad</p> <p>3.3 Listas</p> <p>3.3.1 Definición y representación 3.3.2 Operaciones de inserción y extracción 3.3.3 Implementación dinámica con diferentes tipos de datos 3.3.4 Implementación con arreglos de diferentes tipos de datos 3.3.5 Ligadas, doblemente ligadas, circulares y ordenadas</p> <p>3.4 Árboles</p> <p>3.4.1 Definición y representación 3.4.2 Clasificación, operaciones de inserción, extracción, búsqueda y recorrido</p> <p>3.4.2.1 Árboles Binarios 3.4.2.2 Árboles AVL 3.4.2.3 Árboles B</p> <p>3.4.3 Implementación de árboles con diferentes tipos de datos 3.4.4 Bosques</p>
16	8	4	<p>TÉCNICAS DE ORDENAMIENTO Y BÚSQUEDA</p> <p>Objetivo particular: El alumno aplicará las estructuras de datos en la implementación de algoritmos de ordenación y búsqueda de datos.</p> <p>Temas:</p> <p>4.1 Métodos de ordenación</p> <p>4.1.1 Selección 4.1.2 Inserción (Shell) 4.1.3 Intercambio 4.1.4 Mezcla 4.1.5 Rápida (quick sort) 4.1.6 Polifase 4.1.7 Cascada 4.1.8 Oscilante</p> <p>4.2 Técnicas de búsqueda</p> <p>4.2.1 Comparación de llaves (lineal y binaria) 4.2.2 Transformación de llaves (funciones de Hash y colisiones)</p>

Referencias básicas:

- Bowman, C. (1999). *Algoritmos y estructuras de datos*. México: Oxford.
- Cairo y Guardati. (1999). *Estructuras de datos*. México: McGraw Hill.
- Cairó, Osvaldo & Guardati, Silvia. (2006). *Estructuras de datos* (3 ed). México: McGraw-Hill.
- Guardati, Silvia. (2007). *Estructura de Datos Orientada a Objetos*. México: Pearson.
- Knuth, D. (2002). *El arte de programar ordenadores, Algoritmos fundamentales* (Vol. 1).Barcelona: Reverte.
- Koffman, Elliot B. & Wolfgang, Paul A.T. (2008). *Estructura de datos con C++*. Objetos, abstracciones y diseño. México: McGraw-Hill.
- Kruse, et al. (1999). *Data structures and program design in C++*. E. U. A.: Prentice Hall.
- Kruse, R. (1989). *Programming with data structures*. E.U.A.: Prentice Hall.

- Lipschutz, S. (1987). *Estructura de datos*. México: McGraw Hill.
- Sedgewick, R. (1992). *Algorithms in C++*. E.U.A: Addison Wesley.
- Tenenbaum, et al. (1993). *Estructura de datos en C*. E.U.A.: Prentice Hall.
- Weiss, M. (1992). *Data structures and algorithm analysis*. E.U.A.: Addison Wesley.
- Weiss, Mark Allen (2006). *Data Structures & Problem Solving Using Java* (3ed.). E.U.A.: Addison Wesley.
- Wirth, N. (1992). *Algoritmos + estructuras de datos = programas*. España: Castillo.

Referencias complementarias:

- Aho, et al. (1988). *Estructura de datos y algoritmos*. E.U.A: Addison Wesley.
- Heileman, G. (1997). *Estructura de datos, algoritmos y programación orientada a objetos*. México: McGraw Hill.
- Knuth, D. (1980). *Algoritmos fundamentales*. México: Reverte.
- Wirth, N. (1987). *Algoritmos y estructura de datos*. México: Prentice Hall.

Sugerencias didácticas:	Sugerencias de evaluación del aprendizaje:
Analizar y producir textos Utilizar tecnologías multimedia Resolver ejercicios dentro y fuera de clase Estudiar casos Instrumentar técnicas didácticas como exposición oral, interrogatorio, técnicas grupales, de trabajo colaborativo, trabajo de investigación, entre otros. Realizar visitas de observación. Prácticas de campo Usar recursos didácticos en línea. Implementar algoritmos en un lenguaje de programación.	Examen final oral o escrito Exámenes parciales Informes de prácticas Informes de investigación Participación en clase Rúbricas Solución de ejercicios Trabajos y tareas Proyecto práctico

Perfil Profesiográfico: El profesor que imparta la asignatura deberá tener el título de licenciado en Matemáticas Aplicadas y Computación o carrera afín, con experiencia profesional y docente en la materia, contar con actualización en el área y preferentemente tener estudios de posgrado.