

UNIVERSIDAD AUTONOMA DE BAJA CALIFORNIA

COORDINACIÓN DE FORMACIÓN BÁSICA COORDINACIÓN DE FORMACIÓN PROFESIONAL Y VINCULACIÓN UNIVERSITARIA PROGRAMA DE UNIDADES DE APRENDIZAJE POR COMPETENCIAS

I. DATOS DE IDENTIFICACIÓN

1. Unidad Académica: **Facultad de ciencias**

2. Programa (s) de estudio: (Técnico, Licenciatura): **Licenciatura en Ciencias Computacionales** 3. Vigencia del plan: **2008-1**

4. Nombre de la Unidad de Aprendizaje: **Estructura de Datos y Algoritmos** 5. Clave: 9823

6. HC: 2 HL: 2 HT: 2 HPC: HCL: HE: 2 CR: 8

7. Ciclo Escolar: **2009-1** 8. Etapa de formación a la que pertenece: **Disciplinaria**

9. Carácter de la Unidad de Aprendizaje: **Obligatoria**

10. Requisitos para cursar la Unidad de Aprendizaje: **Introducción a la programación**

Formuló: M. C. Everardo Gutiérrez López

María Victoria Meza Kubo

Fecha: 20 de abril de 2009

Vo. Bo.: Marcelo Rodríguez Meraz

Cargo: Subdirector

II. PROPÓSITO GENERAL DE LA UNIDAD DE APRENDIZAJE

En el ámbito de las ciencias computacionales, es de vital importancia el uso de estructuras que nos permitan almacenar, organizar y representar datos, así como la utilización de los algoritmos que nos ayuden a explotar esas estructuras con la finalidad de realizar un manejo más eficiente de la información. De tal forma que la inclusión de un curso que introduzca al alumnado en estos tópicos es esencial en cualquier programa que pretenda formar profesionales de las ciencias computacionales.

III. COMPETENCIA(S) DE LA UNIDAD DE APRENDIZAJE

Discriminar las estructuras de datos y los algoritmos que faciliten representar y manipular la información necesaria para solucionar problemas específicos.

IV. EVIDENCIA(S) DE DESEMPEÑO

Utilización de estructuras de datos y sus respectivos algoritmos para el manejo de la información en la solución de problemas prácticos.

V. DESARROLLO POR UNIDADES

UNIDAD I. Introducción a las estructuras de datos.

Competencia: Identificar los antecedentes y conceptos generales de las estructuras de datos, su importancia y utilización para representar datos simples aplicados en problemáticas reales.

Contenido temático

Duración

1. INTRODUCCIÓN A LAS ESTRUCTURAS DE DATOS

- 1.1. Propósito del curso y conceptos generales
- 1.2. Representación de tipos de datos básicos: datos binarios, enteros sin signo, enteros con signo, caracteres, punto fijo, punto flotante.
- 1.3. Tiempo de ejecución: concepto general de eficiencia y tiempo de ejecución, manipulación de bits, operaciones aritméticas básicas y su costo en los diferentes tipos de datos.
- 1.4. Programación en pseudocódigo: especificación de un lenguaje en pseudocódigo para la descripción de algoritmos.
- 1.5. Arreglos unidimensionales y bidimensionales: conceptos básicos, manejo en los dispositivos de almacenamiento, aplicación en la solución de problemas prácticos.
- 1.6. Registros: conceptos básicos, manejo en los dispositivos de almacenamiento, aplicación en la solución de problemas prácticos.

UNIDAD II. Estructuras básicas de información.

Competencia: Utilizar las operaciones primitivas de manipulación en las estructuras de datos básicas y aplicarlas en la solución de problemas prácticos.

Contenido temático	Duración
<p>2. ESTRUCTURAS BÁSICAS DE INFORMACIÓN</p> <p>2.1. Pilas: conceptos generales, método de acceso, algoritmos de manipulación (inserción, borrado, etc.), utilización de diferentes tipos de datos, implementación de pilas en arreglos, aplicación en problemas prácticos.</p> <p>2.2. Colas: conceptos generales, métodos de acceso, algoritmos de manipulación (inserción, borrado, etc.), utilización de diferentes tipos de datos, implementación de colas en arreglos, variantes (colas circulares y colas con prioridad), aplicación en problemas prácticos.</p> <p>2.3. Listas ligadas: conceptos generales, apuntadores y memoria dinámica, algoritmos de manipulación (tipos de inserción, tipos de borrado, etc.), utilización de diversos tipos de datos, variantes (listas circulares y listas doblemente ligadas), utilización de listas ligadas para implementar pilas y colas, aplicación en problemas prácticos.</p>	
<p>UNIDAD III. Ordenamiento y búsqueda.</p> <p>Competencia: Discriminar los diferentes algoritmos de ordenamiento y búsqueda, comparando su eficiencia así como sus casos de uso óptimo.</p>	
Contenido temático	Duración
<p>3. ORDENAMIENTO y BÚSQUEDA</p> <p>3.1. Introducción y conceptos generales.</p> <p>3.2. Ordenamiento por intercambio.</p> <p>3.3. Ordenamiento por selección.</p> <p>3.4. Ordenamiento por inserción.</p> <p>3.5. Recursividad y su simulación utilizando pilas.</p> <p>3.6. Ordenamiento por el esquema divide y vencerás.</p> <p>3.7. Ordenamiento utilizando montículos.</p> <p>3.8. Búsqueda secuencial.</p> <p>3.9. Búsqueda binaria (iterativa y recursiva).</p> <p>3.10. Acceso directo.</p>	

UNIDAD III. Árboles.

Competencia: Introducir el manejo de estructuras de árboles y su utilización en los sistemas computacionales.

Contenido temático

Duración

4. ÁRBOLES

4.1. Introducción y conceptos generales.

4.2. Árboles binarios: representación, recorridos.

4.3. Árboles binarios de búsqueda: algoritmos de manipulación (construcción, búsqueda, sucesor, predecesor, inserción, borrado, etc.), variantes (árboles rojo negro).

4.4. Árboles B y su uso en sistemas de almacenamiento de información.

4.5. Códigos de Huffman y su uso en la compresión de datos.

4.6. Grafos: conceptos generales, representación, algoritmos básicos (búsquedas, árbol de expansión mínima, etc.), aplicación en problemas prácticos.

IV. ESTRUCTURA DE LAS PRÁCTICAS

No. de Práctica	Competencia(s)	Descripción	Material de Apoyo	Duración
1	Constatar la forma en que se almacenan los tipos de datos simples y medición práctica de la eficiencia de una función.	Verificar que cada tipo de dato ocupa diferentes espacios de almacenamiento. Medir el tiempo de ejecución de las operaciones aritméticas básicas.	Bibliografía, Computadora, Lenguaje Compilador.	1 sesión
2	Uso de arreglos y registros en la representación y solución de problemas.	Utilizar estructuras de arreglos y registros en un programa básico de manipulación de datos (insertar, borrar, buscar, etc.).	Bibliografía, Computadora, Lenguaje Compilador.	1 sesión
3	Empleo de pilas y colas en soluciones algorítmicas.	Implementar las operaciones básicas de las pilas y colas, y utilizarlas en el desarrollo de una aplicación (manejo de notaciones prefijas, postfijas, colas de espera, etc.)	Computadora, Lenguaje Compilador.	2 sesiones
4	Empleo de listas ligadas en soluciones algorítmicas.	Implementar las operaciones básicas de las listas ligadas y utilizarlas en el desarrollo de una aplicación (simulación de pilas y colas, calendarización de procesos, etc.)	Computadora, Lenguaje Compilador.	1 sesión
5	Discriminar la forma funcional y la eficiencia de los diferentes métodos de ordenamiento y búsqueda.	Utilizar diferentes métodos de ordenamiento y búsqueda en la solución de problemas que manejen grandes cantidades de datos.	Computadora, Lenguaje Compilador.	3 sesiones

7	Empleo de recursividad en soluciones algorítmicas.	Comparar la funcionalidad y eficiencia de soluciones recursivas y sus contrapartes iterativas.	Computadora, Lenguaje Compilador.	1 sesión
8	Uso de árboles para el almacenamiento y acceso a la información.	Implementar las operaciones básicas para el manejo de al menos una estructura de árboles (árboles binarios de búsqueda, árboles rojo-negro, árboles B, etc.).	Computadora, Lenguaje Compilador.	2 sesiones
9	Utilización de estructuras de árbol en una aplicación práctica.	Utilizar las estructuras de árboles para la solución de problemas prácticos (almacenamiento de información, compresión de datos, etc.).	Computadora, Lenguaje Compilador.	2 sesiones

VII. METODOLOGÍA DE TRABAJO

Promover el razonamiento de los temas por parte del alumnado con el objetivo de dejar clara la lógica que sigue cada una de las estructuras, y sus respectivos algoritmos, utilizadas para la representación, almacenamiento y manipulación de los datos. De esta forma los alumnos podrán aplicar los temas del curso en la creación de soluciones a problemas prácticos. Para lograr ese objetivo es de suma importancia que se implementen en un lenguaje de programación moderno las estructuras y los algoritmos que conforman el curso.

VIII. CRITERIOS DE EVALUACIÓN

Criterios propuestos:

Exámenes: 60%

Ejercicios: 40%

Se recomienda incentivar la participación y la ampliación de los ejercicios a través del otorgamiento de puntajes extraordinarios.

IX. BIBLIOGRAFÍA

Básica

- Data Structures and Algorithms
Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman
Addison-Wesley, 1983
- Introduction to Algorithms
Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest,
Clifford Stein
The MIT Press, 2ª Ed., 2001

Complementaria

- Estructura de Datos
Osvaldo Cairó, Silvia Guardati
MacGraw-Hill 1993
- Data Structures using C and C++
Yedidyah Langsam, Moshe J. Augenstein, Aaron M. Tenenbaum
Prentice Hall, 2ª Ed., 1995
- Data Structures and Algorithms in Java
Robert Lafore
Sams, 2ª Ed., 2002
- Estructura de Datos y Diseño de Programas
Robert L. Kruse
Prentice Hall