



UNIVERSIDAD NACIONAL EXPERIMENTAL DEL TACHIRA
VICERECTORADO ACADÉMICO
DEPARTAMENTO DE INGENIERÍA EN INFORMÁTICA

PROGRAMA: ESTRUCTURA DE DATOS

				Especialidad: Ingeniería en Informática			
				Densidad		Horaria	
Código	Nombre de la Asignatura	Estrategias Metodológicas	Unidades Crédito	H.T.	H.P.	T. hrs/sem.	Pre-requisito 8201
8204	Estructura de Datos	Ver anexo	04	4	2	6	

INTRODUCCIÓN

Las Estructuras de Datos constituyen un pilar fundamental en el área de la informática. Estas se emplean para desarrollar aplicaciones de suma importancia tales como sistemas operativos, software de simulación, etc. Algunas de estas son tan fundamentales que le son implantadas físicamente a dispositivos de hardware tales como los microprocesadores. En este curso se orienta al alumno con la finalidad de que conozca cada una de estas estructuras, las compare con analogías de la vida real, las analice desde un punto de vista lógico en el cual sea capaz de representar estas estructuras en forma abstracta para entender sus operaciones de acceso y así posteriormente poder implementar dichas operaciones en un lenguaje de programación de fácil entendimiento para utilizarlas en el diseño de aplicaciones de simulación sencillas.

OBJETIVO GENERAL

Formar alumnos que sean capaces de conocer y manejar las estructuras de datos desde el punto de vista de su aplicación, análisis de las operaciones de acceso, e implementación e algún lenguaje de programación.

DESCRIPCIÓN SINÓPTICA DE LA ASIGNATURA

Introducción a la programación modular y estructurada, arreglos, registros, pilas, colas, listas enlazadas, doblemente enlazadas, listas como pilas, listas como colas, árboles, árboles binarios, recursividad, algoritmos recursivos para árboles, montículos, algoritmos de ordenación y búsqueda, herramientas de programación e introducción a la programación orientada a objetos, clases, objetos, encapsulación, métodos, herencia, polimorfismo, sobrecarga de operadores y funciones, constructores y destructores, clases predefinidas para desarrollar aplicaciones bajo Windows.

UNIDAD I. FUNDAMENTOS DE PROGRAMACIÓN ESTRUCTURADA.			
OBJETIVOS ESPECÍFICOS	CONTENIDOS	ACTIVIDADES	EVALUACION
Al finalizar esta unidad el estudiante debe estar en capacidad de : 1. familiarizarse con el Lenguaje Turbo Pascal. 2. Reconocer los diferentes tipos de datos que manejan los lenguajes de programación. 3. Ajustar la lógica de programación a los lenguajes estructurados y conocer las diferentes instrucciones para la toma de decisiones y realización de ciclos. 4. Aprender a programar de forma modular utilizando métodos imperativos y funcionales para dividir grandes problemas en subproblemas (Divide y Vencerás).	1- Introducción a la programación en Turbo Pascal. 2- Tipos de Datos, Variables y Constantes. 3- Programación Estructurada. 3.1.-Programación Estructurada en Pascal. 3.1.1 .-Estructuras Secuenciales. 3.1.2.-Estructuras de Control Selectivas. 3.1.3.-Estructuras de Control Repetitivas. 4. Programación Modular. 4.1 .-Procedimientos y Funciones. 4.2.-Transferencia de información entre módulos. Los Parámetros. 4.2.1 .-Parámetros Actuales. 4.1.2.-Parámetros Formales. 4.2.2.1 .-Parámetros Valor. 4.2.2.2.-Parámetros Referencia.	Clases Teóricas. Ejercicios.	

UNIDAD II. INTRODUCCIÓN A LAS ESTRUCTURAS DE DATOS			
OBJETIVOS ESPECÍFICOS	CONTENIDOS	ACTIVIDADES	EVALUACION
Al Finalizar Esta Unidad El Estudiante Debe Estar En Capacidad De : 1. Definir El Concepto De Estructura De Datos. 2. Analizar las Estructuras de Datos desde las tres perspectivas fundamentales. 3. Diferenciar las Estructuras de Datos de acuerdo a su origen. 4. Diferenciar las Estructuras de Datos según la asignación de memoria.	1.- Concepto de Estructura de Datos. 2.- Perspectivas de Análisis de una Estructura de datos: 2.1.- Nivel de Aplicación. 2.2.- Nivel Lógico o Abstracto. 2.3.- Nivel de Implementación. 3.- Estructuras de Datos según su origen. 3.1.- Estructuras de Datos Incorporadas. 3.2.- Estructuras de Datos No Incorporadas. 4.- Estructura de Datos según la asignación de memoria: 4.1.- Estructuras Estáticas. 4.2.- Estructuras Dinámicas.	Clases Teóricas.	

UNIDAD III. ESTRUCTURAS DE DATOS ESTÁTICAS.			
OBJETIVOS ESPECÍFICOS	CONTENIDOS	ACTIVIDADES	EVALUACION
Al finalizar esta unidad el estudiante debe estar en capacidad de : 1. Analizar, Implementar y Aplicar las estructuras de datos conocidas como Arreglos y Registros para el desarrollo de aplicaciones. 2. Conocer, Analizar e Implementar la Estructura de Datos PILA, determinar a partir del nivel lógico el conjunto de operaciones que se utilizan para manipular la misma, programarlas y emplearlas en el desarrollo de aplicaciones. 3. Conocer, Analizar e Implementar la Estructura de Datos COLA y los diferentes tipos, determinar a partir del nivel lógico el conjunto de operaciones que se utilizan para manipular la misma, programarlas y emplearlas en el desarrollo de aplicaciones.	1. Arreglos y Registros. 1.1 .-Arreglos Unidimensionales y Bidimensionales. 1.1.1.-Definición. 1.1.2.-Nivel de aplicación. 1.1.3 .-Nivel Lógico. 1.1.4.-Nivel de Implementación. 1.2.-Registros. 1.2.1.-Definición. 1.2.2.-Nivel de aplicación. 1.2.3.-Nivel Lógico. 1.2.4.-Nivel de Implementación. 1.2.5.-Registros Anidados. 1.2.6. -Registros Variantes. 1.3.-Combinación de Arreglos y Registros. 1.3.1 .-Arreglos de Registros. 1.3.2. -Registros con Arreglos. 1.3.3.-Arrglos de Registros con Arreglos. 2. Pilas. 2.1 .-Nivel de Aplicación y Definición. 2.2.-Nivel Lógico. 2.2.1.-Deducción del paquete de operaciones. 2.3.-Nivel de Implementación. 2.3.1.-Desarrollo y codificación de las operaciones necesarias para el manejo de pilas en un lenguaje de programación (Turbo Pascal). 2.4.-Programas de Aplicación.	Clases Teóricas. Ejercicios. Prácticas de Laboratorio. Dinámicas grupales	

	3. Colas. 3.1 .-Nivel de Aplicación y Definición. 3.2.-Tipos de Colas. 3.2.1.-Colas de Frente Fijo. 3.2.2.-Colas de Frente Flotante. 3.2.3.-Colas Circulares. 3.3.-Nivel Lógico para cada uno de los tipos. 3.3.1 .-Deducción de los paquetes de operaciones. 3.4.-Nivel de Implementación para cada uno de los tipos. 3.4.1.-Desarrollo y codificación de las operaciones necesarias para el manejo de pilas en un lenguaje de programación (Turbo Pascal). 3.5.-Programas de Aplicación. 65		
--	--	--	--

UNIDAD IV. ESTRUCTURAS DE DATOS DINÁMICAS.			
OBJETIVOS ESPECÍFICOS	CONTENIDOS	ACTIVIDADES	EVALUACION
Al finalizar esta unidad el estudiante de estar en capacidad de: 1. Desarrollar programas utilizando variables de memoria dinámicas. Conocer como se estructura un programa en la memoria RAM del Computador (Direccionamiento Segmentado de la Intel). 2. Conocer, Analizar e Implementar la Estructura de Datos LISTA y los diferentes tipos, determinar a partir del nivel lógico el conjunto de operaciones que se utilizan para manipular la misma, programarlas y emplearlas en el desarrollo de aplicaciones. Conocer, Analizar e Implementar Árboles de Búsqueda Binaria, determinar a partir del nivel lógico el conjunto de operaciones que se utilizan para manipular los mismos, programarlos y emplearlos en el desarrollo de aplicaciones.	1 . Asignación Dinámica de la Memoria. 1.1 .-Los programas y la memoria, 1.2.-Variables punteros y Variables Dinámicas. 1.2.1.-Declaración. 1.3. -El montículo (Heap). 1.4.-Manejo y operación de los punteros. 1.5.-La constante NIL. 1 .6.-Asignación y liberación de memoria para los punteros. 1.6.1 .-New y Dispose. 1.6.2.-Mark y Reléase. 1.6.3. -Getmem y Freemem. 1.7.-Aspectos que deben tomar en cuenta cuando se utilizan punteros. 2. Listas. 2.1.-Nivel de Aplicación y definición. 2.2.-Tipos de listas. 2.3.-Listas Enlazadas. 2.3.1.-Nivel Lógico. 2.3.2.-Anatomía de un nodo. 2.3.3.-Declaración de la lista. 2.3.4.-Paquete de Operaciones. (Implementado en Pascal). 2.3.5.-Recorrido de la lista. 2.3.6.-Operaciones de Inserción y supresión de un nodo en una lista ordenada. 2.4.-Listas como pilas. 2.5.-Listas como colas. 2.6.-Listas Doblemente Enlazadas. 2.6.1.-Nivel Lógico.	Clases teóricas. Ejercicios. Prácticas de Laboratorio. Dinámicas grupales.	

	2.6.2.- Anatomía de un nodo. 2.6.3. -Paquete de operaciones. (Implementado en Pascal) 2.7.-Programas de Aplicación. 3. Árboles Binarios de Búsqueda. 3.1 .-Nivel de Aplicación y definición. 3.2.-Terminología. 3.3.-Nivel Lógico. 3.4.-Anatomía de un nodo. 3.5.-Declaración del Árbol. 3.6.-Paquete de operaciones. 3.6.1 .-Búsqueda de un elemento. 3.6.2.-Inserción de un nodo. 3.6.3.-Supresión de un nodo. 3.6.3.1 .-Supresión de un hoja. 3.6.3.2.-Supresión de un nodo con un solo hijo 3.6.3.3.-Supresión de un nodo con dos hijos. 3.7.-Recorridos de un Árbol. 3.7.1.-Recorrido In Orden.		2do Parcial 30% Asignación del Proyecto
--	--	--	---

UNIDAD V. TÉCNICAS AVANZADAS DE PROGRAMACIÓN.			
OBJETIVOS ESPECÍFICOS	CONTENIDOS	ACTIVIDADES	EVALUACION
Al finalizar esta unidad el estudiante debe estar en capacidad de: 1. Comprender la recursividad y analizar sus ventajas y desventajas, para determinar cuando es adecuado implementar una rutina recursiva. 2. Ejecutar en frío algoritmos recursivos clásicos. 3. Diseñar algoritmos recursivos para la solución de problemas. 4. Recorrer Arboles binarios de búsqueda recursivamente. 5. Aplicar métodos de ordenación y búsqueda recursivos y no recursivos. 6. Iniciarse en la programación orientada a objetos.	1. Recursividad. 1.1.-Definición. 1.2.-Análisis de un problema recursivo. 1.3.-Algoritmos recursivos clásicos. 2. Recorridos Recursivos de un Árbol Binario de Búsqueda. 2.1 .-Recorrido In Orden. 2.2.-Recorrido Pre Orden. 2.3.-Recorrido Post Orden. 3. Ordenación y Búsqueda. 3.1.-Ordenación. 3.1.1 .-Ordenación por el método de la Burbuja. 3.1.2.-Ordenación por Selección. 3.1.3 .-Ordenación por Inserción. 3.1.4.-Ordenación Rápida (Quicksort). 3.2.-Búsqueda: 3.2.1 .-Búsqueda Lineal. 3.2.2.-Búsqueda Binaria. 3.2.3.-Búsqueda Binaria Recursiva. 4. Programación Orientada a Objetos en C++. 4.1.- Definición. 4.2.- Objetos. 4.3.- Clases. 4.4.- Métodos. 4.5.-Polimorfismo. 4.6.- Herencia. 4.7.- Fundamentos de C++. 4.8.- Sobrecarga de funciones. 4.9.- Sobrecarga de Operadores. 4.10.-Constructores y Destructores. 4.11 .-Funciones Amigas.	- Clases en Laboratorio Prác y Teoría. -Asignaciones para desarrollar en el Computador.	

	4.12.-Paso de Objetos a Funciones. 4.13. -Arrays de Objetos. 4.14.-Punteros a Objetos. 4.15.-Funciones y Clases Virtuales. 4.16.-OOP bajo Windows. 4.17.-Clase Predefinidas en el Borland CC++ 4.5 para Windows. 4.18.-Desarrollo de Aplicaciones..		3er Parcial 30% Entrega de Proyecto 15%
--	--	--	--

BIBLIOGRAFÍA

- Susan Lily, Nell Dale. Estructuras de Datos y Programación en Pascal.
- Cairo / Guardatti. Estructuras de Datos.
- Tanenbaum. Estructuras de Datos en C.
- H. Schildt. Manual de Referencia C/C++.
- AmushCraig. Aprendiendo Borland C++ en 21 días.
- Luis Joyanes A. Programación Orientada a Objetos.