



PROGRAMACIÓN

Especialista en contenido:	ING. MSC. JESÚS M. ARAQUE	AUTORIZADO POR VICE RECTORADO ACADÉMICO (FIRMA Y SELLO)
Fecha de elaboración:	MARZO, 1991	
Elaborado por:	ING. MSC. JESÚS M. ARAQUE	

DESCRIPCIÓN DEL PROGRAMA

El programa cubre los temas fundamentales para el inicio de estudio de programación en lenguaje PASCAL. El programa incluye tres unidades:

1. VALIDACIÓN
2. MODULARIDAD
3. MANEJO DE ESTRUCTURA DE DATOS

MODALIDAD Y ESTRATEGIA DE ENSEÑANZA

Conforme a la naturaleza del curso y a las modalidades y estilos de enseñanza sugeridos por la UFT, el docente empleará las siguientes modalidades y estrategias de enseñanza. Revisión de la literatura/controles de lectura. Explicaciones Teórico-prácticas. Ejercicios de Desempeño. Asignación de Trabajos.

ESTRATEGIA DE ESTUDIO

.Se recomienda una sólida preparación y revisión previa a cada sesión de aquellos temas que serán tratados.

Además de las tareas e estudio individual, se sugiere formar grupos de estudio estables para intercambiar notas y opiniones, y para reforzar colectivamente las destrezas conceptuales y prácticas. De máxima importancia es una constante dedicación por la resolución de ejercicios propuestos en clase así como el uso del computador para correr los programas codificados en PASCAL

OBJETIVO TERMINAL DE LA ENSEÑANZA

Desarrollar en el estudiante la habilidad para resolver en forma apropiada problemas científicos mediante el lenguaje de programación PASCAL

UNIDAD I	OBJETIVO TERMINAL DE UNIDAD
VALIDACIÓN	AL FINALIZAR ESTA UNIDAD, EL ESTUDIANTE SERÁ CAPAZ DE DESARROLLAR PROGRAMAS DE COMPUTACIÓN DE MEDIANA COMPLEJIDAD PARA APLICACIONES COMERCIALES Y CIENTÍFICAS, QUE SEAN CLAROS, CORRECTOS, COMPLETOS Y MODULARES.
DURACIÓN	
5 SEMANAS	
EVALUACIÓN	
ESTRATEGIA DE INSTRUCCIÓN	
Revisión de literatura/controles de lectura. Explicaciones teórico-prácticas. Ejercicios de Desempeño. Asignaciones de Trabajos.	
OBJETIVOS ESPECÍFICOS	
Presentada una serie de conceptos y en base a discusiones dirigidas y tareas asignadas en clase: - EXPRESAR en forma clara las características de un buen programa. - VALIDAR datos de entrada en programas interactivos. - VALIDAR datos en programas con procesamiento en lote. - GENERAR archivos de salida de datos y de resultaos. - PROCESAR archivos de datos carácter a carácter. - ELABORAR datos de pruebas para programas.	
CONTENIDOS	
- Características de un buen programa: claro, correcto, completo y modular (ejemplos ilustrativos) - Validación de datos de entrada en programas interactivos: char, una tecla espera. Entero o real, dentro del rango esperado. Fin de datos, uso de centinela, pregunta de fin de proceso, datos que se generan. (Expresiones aritméticas y lógicas, operadores asociado; estructuras de control, read (In) Write (In), unidad de turbo Pascal. - Validación de datos en programas con procesamiento en lote. - Generación de archivos de salida de datos y de resultados. (Eof, manejo de errores de E/S en el procesamiento de archivos externos) - Procesamiento de archivos de texto carácter a carácter. (Ecf, tratamiento de los dispositivos de E/S como archivos) - Elaboración de datos de prueba para programas.	

UNIDAD II		OBJETIVO TERMINAL DE UNIDAD
MODULARIDAD		DEFINIDO UN PROBLEMA EL ESTUDIANTE SERÁ CAPAZ DE DISEÑAR, IMPLEMENTAR Y PROBAR PROGRAMAS DE MEDIANA COMPLEJIDAD EN FORMA COMPLETAMENTE MODULAR.
DURACIÓN		
5 SEMANAS		
EVALUACIÓN		
ESTRATEGIA DE INSTRUCCIÓN		
Revisión de literatura/controles de lectura. Explicaciones teórico-prácticas. Ejercicios de Desempeño. Asignaciones de Trabajos.		
OBJETIVOS ESPECÍFICOS		
- JUSTIFICAR funciones y procedimientos - DEFINIR funciones y procedimientos - ACTIVAR funciones y procedimientos - DISEÑAR programas en forma modular - PROBAR subprogramas - USAR unidades del Turbo y otras disponibles - DESARROLLAR unidades		
CONTENIDOS		
- Justificación, definición y activación de funciones. - Justificación, definición y activación de procedimientos. - Portabilidad de los subprogramas. - Parámetros de valor y de referencia. - Alcance de variables. - Diseño Modular de programas y prueba modular de subprogramas (uso del Stubs y de Drivers) - Uso de las unidades del Turbo y de las unidades disponibles. (Dos, Hierros, Ventanal, etc.) Desarrollo de unidades.		

UNIDAD III		OBJETIVO TERMINAL DE UNIDAD
MANEJO DE ESTRUCTURA DE DATOS		AL FINALIZAR ESTA UNIDAD, EL ESTUDIANTE SERÁ CAPAZ DE SELECCIONAR LAS ESTRUCTURAS DE DATOS ADECUADAS A UN PROBLEMA DADO Y HACER UN MANEJO EFICIENTE DE LAS MISMAS. PARA ELLO, IMPLEMENTAR LOS SUBPROGRAMAS NECESARIOS, O BIEN, USAR LAS FACILIDADES PROVISTAS POR EL LENGUAJE PARA CADA UNA DE ESAS ESTRUCTURAS DE DATOS.
DURACIÓN		
7 SEMANAS		
EVALUACIÓN		
ESTRATEGIA DE INSTRUCCIÓN		
Revisión de literatura/controles de lectura. Explicaciones teórico-prácticas. Ejercicios de Desempeño. Asignaciones de Trabajos.		
OBJETIVOS ESPECÍFICOS		
- JUSTIFICAR, DEFINIR y MANIPULAR arreglos - SELECCIONAR las estructuras de datos adecuadas a un problema dado - MANEJAR eficientemente las estructuras de datos - GENERAR diversos índices para manipular arreglos de registros - ADAPTAR algoritmos de búsqueda en arreglos, a algoritmos de búsqueda en un archivo - SIMULAR búsqueda secuencial en un archivo		
CONTENIDOS		
- Arreglos: Justificación, definición y manipulación. - Enumerados y subrangos. Funciones para enumerados. Enumerados como subíndice de arreglos. - Algoritmos portables de búsqueda y de ordenamiento en arreglos. (Arreglos con coordenadas, (Fila columna), significativas. Problemas de Categorización. Impresión de Diagramas de barras verticales). - Arreglos multidimensionales. - Procedimientos y funciones para el procesamiento de STRINGS. - Concepto y definición de conjuntos. - Operaciones de conjuntos. (Simulación de conjuntos con arreglos) - Requisitos, conceptos, definición y representación interna. Registros Jerárquicos. - Arreglos de Registros. Algoritmos portables para su manipulación. (Generación de diversos índices para manipulación de arreglos de registros. Adaptación de algoritmos de búsqueda en arreglos, a algoritmos de búsqueda en un archivo. Simulación de búsqueda secuencial en un archivo.		

BIBLIOGRAFÍA

Alonso y Morales (1988). Problemas de Programación, Programación Estructurada. Paraninfo.

_____ (1988). Problemas de Programación. Paraninfo.

Bowles Kenneth L. Franklin Stephen D. Volper Dennis (1987). Soluciones de Problemas con Pascal UCSD. Limusa.

C. Galan (1988). Pascal y Turbo Pascal. Paraninfo.

Coñat, Andrés y Nuñez, Sara (1988). Turo Pascal Práctico y Programas. Paraninfo.

Crawley Winston. McArthur William. Pascal. Programación Estructurada.

Dale Nell. Weems Chip. (1989). Pascal. Segunda Edición. McGraw Hill.

Konvaline, Kris y Nameroff, Steven. (1989). Programación Tools in Pascal. McGraw Hill

Jamsa, Kris y Nameroff, Steven (1988). Turbo Pascal. Biblioteca de Programa McGraw Hill

Joyanes, A. Luis (1990). Programación en Turbo Pascal. McGeraw Hill

Koffman Elliot B. Pascal. Introducción al Lenguaje y Resolución de Problemas con Pascal UCSD.

_____ (1987). Turbo Pascal. Editorial Iberoamericana.

Schneider G. Michael, Weingart Steeven W., Perlman David (1986). Introducción a la Programación y Solución de Problemas con Pascal. Limusa.

Welsh Jim, Elder John (1983). Pascal Introducción. Prentice Hall.

Zaks, Rodnay (1987). Turbo Pascal. Editorial Iberoamericana.

Yester, M. (1989). Using Turbo Pascal (Versión 5.0) Que Corp