

PRESENTACION

El proceso evolutivo de la sociedad ha alcanzado nuestros ámbitos universitarios de tal forma que nuestra Facultad ha visto en los últimos tiempos el proceso de rediseño curricular.

Al respecto, el Ing. Guadalupe Cedillo Garza, director actual, ha mencionado: "este proceso es de gran importancia para la FACULTAD DE INGENIERIA MECANICA Y ELECTRICA; es también importante que maestros y alumnos participen en este proceso de actualización, ahora ya de carácter permanente, en justificación principal radica en el compromiso universitario de formar excelentes egresados ávidos de asimilar las innovaciones científicas y tecnológicas, dispuestos a aportar su esfuerzo en una forma responsable al sector productivo y a la sociedad en general."

Conciente de las necesidades que involucra este gran cambio, la Coordinación de Administración y de Sistemas modificó sus programas de clase, entre ellos el de la clase de Análisis Numérico e introdujo el lenguaje de programación Pascal para microcomputadoras.

Como un esfuerzo de apoyo a dichas modificaciones, la coordinación y un servidor presentamos el siguiente texto a la comunidad universitaria el cual fue procesado en tipografía LASER en el Centro de Diseño Y Mantenimiento de Instrumentos de la Facultad de Ingeniería Mecánica Y Eléctrica.



Ing. F. Eugenio Guerrero

FONDO UNIVERSITARIO Diciembre

37740

INTRODUCCION

CONTENIDO	PAGINA
INTRODUCCION	1
CAPITULO 1.- CONCEPTOS BASICOS	2
INTRODUCCION INFORMAL AL PASCAL	2
ESTRUCTURA DEL PROGRAMA	3
PROCESO COMPILACION/EJECUCION	9
CAPITULO 2.- TIPOS DE DATOS	16
IDENTIFICADORES, LITERALES Y CONSTANTES	17
DATOS	19
TIPO CHAR	19
TIPO INTEGER	20
TIPO REAL	20
TIPO BOOLEAN	20
TIPOS ENUMERADOS	26
CAPITULO 3.- ESTATUTOS DE CONTROL	32
OPERADORES	32
DECLARACION Y EVALUACION DE EXPRESIONES	
SENTENCIAS SIMPLES Y COMPUESTAS	38
DECISIONES	40
CICLOS	45
CAPITULO 4.- PROCEDIMIENTOS Y FUNCIONES	47
DECLARACIONES	50
PARAMETROS	50
CAPITULO 5.- ARREGLOS	51
UNIDIMENSIONALES (VECTORES)	51
BIDIMENSIONALES (MATRICES)	55
CAPITULO 6.- ARCHIVOS	62
ARCHIVOS DE TEXTO	62

APENDICE A.- REFERENCIA COMPLETA DE FUNCIONES	67
APENDICE B.- PALABRAS RESERVADAS	99
APENDICE C.- MENSAJES Y CODIGOS DE ERROR	102
APENDICE D.- COMANDOS DEL EDITOR	120
ESTRUCTURA DEL PROGRAMA	
PROCESO DE COMPILACION	
CAPITULO 2 - TIPOS DE DATOS	
IDENTIFICADORES, LITERALES Y CONSTANTES	
DATOS	
TIPO CHAR	
TIPO INTEGER	
TIPO REAL	
TIPO BOOLEAN	
TIPOS ENUMERADOS	
CAPITULO 3 - ESTATUTOS DE CONTROL	
OPERADORES	
DECLARACION Y EVALUACION DE EXPRESIONES	
SENTENCIAS SIMPLES Y COMPUSTAS	
DECISIONES	
CICLOS	
CAPITULO 4 - PROCEDIMIENTOS Y FUNCIONES	
DECLARACIONES	
PARAMETROS	
CAPITULO 5 - ARRAYS	
UNIDIMENSIONALES (VECTORES)	
BIDIMENSIONALES (MATRICES)	
CAPITULO 6 - ARCHIVOS	
ARCHIVOS DE TEXTO	

INTRODUCCION

Pascal fue desarrollado en 1971 por Niklaus Wirth, en respuesta a la creciente necesidad de un lenguaje de programación modular, sistemático y fuertemente implementado en tipos. Wirth, quien había desarrollado anteriormente los lenguajes ALGOL y PL/1, entendía la importancia de proveer un lenguaje que diese una forma de programación estructurada, ayudase al desarrollo de sistemas y facilitara el aprendizaje de tópicos y conceptos avanzados de programación, tales como apuntadores, registros y variables dinámicas.

En 1971, Niklaus Wirth escribió la definición del lenguaje de Pascal, en su bien conocido (e imposible de leer) libro titulado "Pascal: User Manual and Report". El libro contiene una descripción meticulosamente detallada de lo que el lenguaje debe ser. Sin embargo, no existe actualmente un compilador que tome un programa fuente y produzca código ejecutable a partir de él. Algunos (muchos algunos, actualmente), basándose en el libro de Wirth, elaboraron compiladores que han dejado muy atrás su definición de lenguaje. Tales programas aparecieron primero para computadoras de tipo "mainframe", como la CDC6600, y hace algunos años para microcomputadoras.

Esos compiladores, que funcionalmente siguen el cuerpo de la definición de lenguaje de Pascal, se llaman "implementaciones" del lenguaje Pascal.

La definición formal de Pascal, como la hizo Wirth, tiene algunas "lagunas" y partes débiles, con un uso bastante limitado. La definición de Wirth no maneja strings, archivos random, llamadas al sistema operativo, ni muchas otras cosas más. La gente que implementa el lenguaje en una computadora dada, generalmente va mas allá de la definición del lenguaje, haciendo al compilador capaz de compilar programas más útiles.

Además, algunas implementaciones imponen límites al programador, que la definición de Wirth no tiene. Para poder hacer uso de sistemas de memoria pequeña, los límites del mundo real deben forzarse al tamaño de los identificadores, de las funciones y procedimientos, de los números enteros y reales máximos, etc. Ninguno de ellos está presente en la definición del lenguaje, pero son críticos en alguna implementación en particular.

Pascal, por diseño, es un lenguaje estructurado. A diferencia de BASIC y FORTRAN, Pascal impone una estructura en sus programas. No permite estatutos colocados al azar, aun si se toma cada uno y éstos son sintácticamente correctos. Hay un plan maestro detallado que todo programa de Pascal debe seguir. Un programa debe ser codificado en ciertas partes. Algunas deben estar ahí y otras aquí. Todo debe estar en cierto orden. Algunas cosas no pueden trabajar juntas y otras sí. Haciendo a un lado ciertas concesiones que los diseñadores de los compiladores han hecho (Pascal hace esas tareas fácilmente), la estructura de Pascal existe para reforzar cierta forma de pensar acerca de la programación. Esta forma de pensar, desarrollada por Wirth y otros, se denomina "programación estructurada". Representa el énfasis de Niklaus Wirth sobre la creación de programas que sean comprensibles, sin necesidad de páginas de diagramas de flujo y cientos de líneas de explicación. Aunque la programación estructurada puede estar acompañada de casi cualquier lenguaje (inclusive BASIC), Pascal es uno de los pocos que la requiere.

Desde entonces, Pascal ha sido aceptado como estándar para enseñar estructuras de datos y conceptos de programación en muchas universidades. Los graduados se han encargado de llevar el Pascal al resto del mundo. Gracias al TURBO Pascal de la casa Borland, Pascal es el lenguaje de programación para microcomputadoras más ampliamente usado. De hecho, el ambiente de TURBO Pascal es tan popular, que muchos desarrolladores de sistemas no han tenido elección para emigrar al ambiente de programación interactivo que éste provee. En adición, TURBO Pascal ha tenido un impacto significativo en el ISO Pascal estándar. Los compiladores de Pascal de todos los niveles -desde computadoras mainframe hasta micros- tienen mayor inclinación para el Pascal, ya que éste cuenta con un desarrollo de programas interactivo y una poderosa colección de rutinas de librerías. En los años que vienen, los negocios y las organizaciones verán las ventajas de un programador adiestrado en Pascal -especialmente TURBO Pascal- como herramienta para el desarrollo.