

# CONTENIDO

---

Prefacio	xiii
----------	------

## PARTE I **Conceptos preliminares**

---

<b>0</b>	<b>Introducción</b>	<b>2</b>
0.0	En este capítulo	3
0.1	Resolución de problemas <i>Herramientas matemáticas / Niveles conceptuales y de implementación</i>	3
0.2	Paradigmas del lenguaje <i>Paradigmas imperativos / Paradigmas declarativos</i>	6
0.3	Consideraciones prácticas <i>Desde el bajo, pasando por el alto, hasta el muy alto nivel / Programación a gran escala / Problemas especiales</i>	12
0.4	Criterios de lenguaje <i>Descripciones bien definidas / Comprobabilidad / Confiabilidad / Traducción rápida / Código objeto eficiente / Ortogonalidad / Generalidad / Consistencia y notaciones comunes / Uniformidad / Subconjuntos / Extensibilidad / Transportabilidad</i>	16
0.5	Resumen	28
0.6	Notas sobre las referencias	29
<b>1</b>	<b>Variables y tipos de datos</b>	<b>30</b>
1.0	En este capítulo	31
1.1	Tipos de datos primitivos <i>Entero (integer) / Real / Carácter / Booleano / Apuntador</i>	32
1.2	Variables <i>Identificadores / Palabras reservadas y palabras clave / Ligadura / Bloques y alcance / Registros de activación</i>	39
1.3	Tipos de datos estructurados <i>Tipos definidos por el usuario / Tipos agregados / Cuestiones de tipo</i>	51
1.4	Resumen	67
1.5	Notas sobre las referencias	68

McGraw-Hill Interamericana, S.A.  
Eduardo Pérez Dávalos  
COORDINADOR - BOLIVIA  
TEL. FAX (02) 341618-01291250

<b>2</b>	<b>Abstracción</b>	<b>70</b>
2.0	En este capítulo	72
2.1	Abstracción de datos	72
	<i>Los datos y el almacenamiento / Tipos de datos abstractos / Tipos genéricos</i>	
2.2	Abstracción de control	83
	<i>Ramificación / Iteración / Recursión / Excepciones</i>	
2.3	Abstracción de procedimientos	93
	<i>Procedimientos / Módulos y ADT / Clases de ADT / Objetos / Ejecución concurrente</i>	
2.4	Resumen	105
2.5	Notas sobre las referencias	106

## PARTE II **Lenguajes imperativos**

---

<b>3</b>	<b>Estructura en bloques</b>	<b>108</b>
3.0	En este capítulo	110
3.1	ALGOL 60	111
	<i>Viñeta histórica: Diseño por comité / Conceptos de ALGOL 60 / Puntos problemáticos en ALGOL 60 / Especificación del lenguaje</i>	
3.2	ALGOL 68	123
3.3	Pascal	124
	<i>Viñeta histórica: Pascal y Modula-2: Niklaus Wirth / Filosofía y estructura</i>	
3.4	Ada	129
	<i>Viñeta histórica: Ada / Organización del programa / Tipos / La facilidad genérica / Excepciones / El entorno de soporte para programación en Ada (APSE)</i>	
3.5	C	145
	<i>Viñeta histórica: El dúo dinámico: Dennis Ritchie y Kenneth Thompson / Tipos de datos en C / Operadores de C / C y UNIX / El C estándar / Ventajas y desventajas</i>	
3.6	Resumen	160
3.7	Notas sobre las referencias	161
<b>4</b>	<b>Lenguajes para programación orientada a objetos (POO)</b>	<b>164</b>
4.0	En este capítulo	166

4.1	Programación con objetos	167
	<i>Mensajes, métodos y encapsulamiento / Primeras nociones de objetos en Simula / Objetos en Ada 83 y Ada 95</i>	
4.2	Clases y polimorfismo	180
	<i>Procedimientos y paquetes genéricos en Ada / Clases en Object Pascal / Clases en C++ / Implementación de clases heredadas</i>	
4.3	Smalltalk	194
	<i>Viñeta histórica: Smalltalk: Alan Kay</i>	
4.4	Herencia y orientación a objetos	196
	<i>Tipos y subtipos en jerarquías de herencia / Herencia múltiple / Ejemplares de lenguaje / Ligadura dinámica</i>	
4.5	Java	217
	<i>Construcciones del lenguaje Java / Las Interfaces de Programación para Aplicaciones de Java (APIs) / Compilación y ejecución de un programa Java / HotJava y Applets / Tipos de programa / Diferencias entre Java, C y C++</i>	
4.6	Resumen	228
4.7	Notas sobre las referencias	229
<b>5</b>	<b>Construcciones de lenguajes para procesamiento en paralelo</b>	<b>232</b>
5.0	En este capítulo	233
5.1	El paradigma	234
5.2	Procesos múltiples	236
5.3	Sincronización de procesos cooperativos	238
	<i>Semáforos / Monitores / Rendezvous (Punto de reunión) / Paso de mensajes</i>	
5.4	Algunas soluciones de sincronización	247
	<i>Semáforos en ALGOL 68, C y Pascal S / Tipos de proceso y monitor en Concurrent Pascal / Rendezvous (Punto de reunión) en Ada y Concurrent C / Paso de mensajes en Occam</i>	
5.5	Tuplas y objetos	262
	<i>El espacio de tuplas de Linda / Objetos como unidades de paralelismo</i>	
5.6	Administración de fallas parciales	267
5.7	Resumen	268
5.8	Notas sobre las referencias	269

## PARTE III **Autómatas y lenguajes formales**

---

<b>6</b>	<b>Lenguajes formales</b>	<b>272</b>
6.0	En este capítulo	273
6.1	Lenguajes formales	274
	<i>Definición de lenguajes formales / La jerarquía de Chomsky de los lenguajes formales / Viñeta histórica: Clasificaciones de los lenguajes: Noam Chomsky / Viñeta histórica: Alan Turing: Lo que las máquinas no pueden hacer</i>	
6.2	Gramáticas regulares	290
	<i>Expresiones regulares / Autómatas finitos (FA, NFA y DFA) / Aplicaciones</i>	
6.3	Gramáticas Libres de Contexto (CFG)	298
	<i>Autómatas descendentes (PDA; Push-Down Automata) / Árboles de análisis sintáctico / Gramáticas ambiguas / Aplicaciones / Formas normales</i>	
6.4	Gramáticas para los lenguajes naturales	311
6.5	Resumen	313
6.6	Notas sobre las referencias	314

## PARTE IV **Lenguajes declarativos**

---

<b>7</b>	<b>Programación lógica</b>	<b>318</b>
7.0	En este capítulo	319
7.1	Sistemas lógicos formales	320
	<i>Viñeta histórica: Aristóteles / Demostraciones o pruebas / Búsqueda</i>	
7.2	PROLOG	337
	<i>Viñeta histórica: PROLOG: Colmerauer y Roussel / Conversando en PROLOG: hechos, reglas y consultas / Implementaciones de PROLOG / Aplicaciones / Fortalezas y debilidades</i>	
7.3	Resumen	359
7.4	Notas sobre las referencias	361
<b>8</b>	<b>Programación funcional (aplicativa)</b>	<b>362</b>
8.0	En este capítulo	365
8.1	Características de los lenguajes funcionales	365
	<i>Composición de funciones / Funciones como objetos de primera clase / Ausencia de efectos colaterales / Semántica limpia</i>	

8.2	LISP	369
	<i>Viñeta histórica: LISP: John McCarthy / El lenguaje LISP (dialecto SCHEME) / Otras características no funcionales / Dialectos / Common LISP</i>	
8.3	Implementación de lenguajes funcionales	396
	<i>Evaluación débil (lazy evaluation) contra evaluación estricta (strict evaluation) / Alcance y ligaduras / Recolección de basura</i>	
8.4	Soporte de paralelismo con funciones	405
8.5	Otros lenguajes funcionales	407
	<i>APL / ML / Otros</i>	
8.6	Resumen	417
8.7	Notas sobre las referencias	418
<b>9</b>	<b>Lenguajes para bases de datos</b>	<b>420</b>
9.0	En este capítulo	422
9.1	Modelos jerárquicos y de red	422
9.2	El modelo relacional	424
	<i>Manipulación de bases de datos relacionales / SQL / Sistemas basados en lógica utilizando PROLOG</i>	
9.3	Modelos de datos semánticos	434
9.4	Modelo de base de datos orientado a objetos	437
9.5	Resumen	438
9.6	Notas sobre las referencias	439
	Apéndice A	441
	Cálculos lógicos (para el capítulo 7)	
	Apéndice B	451
	El cálculo lambda (para el capítulo 8)	
	Apéndice C	457
	Fuentes de software	
	Referencias	461
	Índice	475

## Laboratorios

---

2.1	Tipos de datos abstractos: Ada/Pascal	80
2.2	Métodos de paso de parámetros: Pascal	101
3.1	Bloques: Ada/Pascal	145
3.2	Combinación de características de bajo y alto nivel: C	157
3.3	Diversión con trucos para C: C	158
3.4	Herramientas IDE: Pascal/C	160
3.5	Herramientas APSE: Ada	160
4.1:	Objetos, encapsulamiento y métodos:	
	Object Pascal/Ada/C++	179
4.2	Polimorfismo: Object Pascal/Ada/C++	193
4.3	Clases y herencia: Object Pascal y C++	216

4.4	Objetos y programación orientada a objetos: Java	228
4.5	HTML para utilizarse en el World Wide Web con Java	228
4.6	Un applet: Java	228
5.1	Simulación de procesamiento en paralelo: Ada	259
5.2	Productores-consumidores: Pascal S/Occam 2/ C-Linda	266
6.1	Expresiones regulares: grep	296
6.2	EBNF: Papel y lápiz	309
7.1	Introducción al lenguaje: PROLOG	334
7.2	Caníbales y misioneros: PROLOG	356
8.1	Practicando con LISP: SCHEME	388
8.2	Una función de palíndromos: SCHEME	388
8.3	Programación utilizando ciclos iterativos: SCHEME	389
8.4	Rastreo y depuración: SCHEME	389
8.5	Programación en SCOOPS: SCHEME	393
9.1	SQL: dBASE IV	433