

---

---

# Contenido

<b>Parte I. Conceptos</b>	<b>1</b>
<b>1 El estudio de los lenguajes de programación</b>	<b>2</b>
1.1 ¿Por qué estudiar lenguajes de programación?	2
1.2 Breve historia de los lenguajes de programación	5
1.2.1 Desarrollo de los primeros lenguajes	5
1.2.2 El papel de los lenguajes de programación	9
1.3 ¿Qué es lo que constituye un buen lenguaje?	12
1.3.1 Atributos de un buen lenguaje	12
1.3.2 Dominios de aplicación	16
1.3.3 Estandarización de lenguajes	19
1.4 Efectos de los entornos sobre los lenguajes	23
1.4.1 Entornos de procesamiento por lotes	23
1.4.2 Entornos interactivos	24
1.4.3 Entornos de sistemas incrustados	25
1.4.4 Entornos de programación	26
1.4.5 Marcos de ambiente	30
1.5 Lecturas adicionales sugeridas	30
1.6 Problemas	31
<b>2 Cuestiones de diseño de lenguajes</b>	<b>33</b>
2.1 La estructura y operación de una computadora	33
2.1.1 El equipo físico (hardware) de la computadora	35
2.1.2 Computadoras de firmware	39
2.1.3 Traductores y computadoras simuladas por <i>software</i>	41
2.2 Computadoras virtuales y tiempos de enlace	45

2.2.1	Sintaxis y semántica	46
2.2.2	Computadoras virtuales e implementación de lenguajes	47
2.2.3	Jerarquías de computadoras	48
2.2.4	Enlace y tiempo de enlace	50
2.3	Paradigmas de lenguajes	55
2.4	Lecturas adicionales sugeridas	59
2.5	Problemas	59
<b>3</b>	<b>Problemas de traducción de lenguajes</b>	<b>61</b>
3.1	Sintaxis de lenguajes de programación	61
3.1.1	Criterios generales de sintaxis	62
3.1.2	Elementos sintácticos de un lenguaje	66
3.1.3	Estructura de conjunto de programas y subprogramas	69
3.2	Etapas de traducción	72
3.2.1	Análisis del programa fuente	74
3.2.2	Síntesis del programa objeto	77
3.3	Modelos formales de traducción	79
3.3.1	Gramáticas BNF	80
3.3.2	Autómatas de estados finitos	89
3.3.3	Autómatas de desplazamiento descendente	93
3.3.4	Algoritmos de análisis sintáctico eficiente	95
3.3.5	Modelado semántico	98
3.4	Lecturas adicionales sugeridas	102
3.5	Problemas	103
<b>4</b>	<b>Tipos de datos</b>	<b>107</b>
4.1	Propiedades de tipos y objetos	107
4.1.1	Objetos de datos, variables y constantes	107
4.1.2	Tipos de datos	112
4.1.3	Especificación de tipos elementales de datos	113
4.1.4	Implementación de tipos elementales de datos	117
4.1.5	Declaraciones	119
4.1.6	Verificación y conversión de tipos	121
4.1.7	Asignación e inicialización	127
4.2	Tipos elementales de datos	130
4.2.1	Tipos de datos numéricos	130
4.2.2	Enumeraciones	137
4.2.3	Booleanos	139

---

4.2.4	Caracteres	140
4.2.5	Internacionalización	141
4.3	Tipos de datos estructurados	142
4.3.1	Objetos de datos estructurados y tipos de datos	142
4.3.2	Especificación de tipos de estructuras de datos	143
4.3.3	Implementación de tipos de estructuras de datos	145
4.3.4	Declaraciones y verificación de tipos para estructuras de datos	149
4.3.5	Vectores y arreglos	151
4.3.6	Registros	160
4.3.7	Listas	167
4.3.8	Cadenas de caracteres	172
4.3.9	Apuntadores y objetos de datos construidos por programador	175
4.3.10	Conjuntos	178
4.3.11	Objetos de datos ejecutables	181
4.3.12	Archivos y Entrada-Salida	181
4.4	Lecturas adicionales sugeridas	187
4.5	Problemas	188
<b>5</b>	<b>Abstracción I: Encapsulamiento</b>	<b>195</b>
5.1	Tipos de datos abstractos	196
5.1.1	Evolución del concepto de tipo de datos	197
5.1.2	Ocultamiento de información	198
5.2	Encapsulamiento por subprogramas	200
5.2.1	Los subprogramas como operaciones abstractas	200
5.2.2	Definición e invocación de subprogramas	203
5.2.3	Definiciones de subprograma como objetos de datos	208
5.3	Definiciones de tipo	209
5.3.1	Equivalencia de tipos	211
5.3.2	Definiciones de tipo con parámetros	215
5.4	Gestión de almacenamiento	216
5.4.1	Elementos principales en tiempo de ejecución que requieren almacenamiento	217
5.4.2	Gestión de almacenamiento controlada por el programador y por el sistema	219
5.4.3	Gestión de almacenamiento estático	220
5.4.4	Gestión de almacenamiento con base en pilas	221

5.4.5	Gestión de almacenamiento en montículos: Elementos de tamaño fijo	223
5.4.6	Gestión de almacenamiento en montículos: Elementos de tamaño variable	231
5.5	Lecturas adicionales sugeridas	234
5.6	Problemas	234
<b>6</b>	<b>Control de secuencia</b>	<b>238</b>
6.1	Control implícito y explícito de secuencia	238
6.2	Secuenciamiento con expresiones aritméticas	239
6.2.1	Representación de estructura de árbol	240
6.2.2	Representación en tiempo de ejecución	248
6.3	Secuenciamiento con expresiones no aritméticas	253
6.3.1	Concordancia de patrones	253
6.3.2	Unificación	257
6.3.3	Retroceso	263
6.4	Control de secuencia entre enunciados	264
6.4.1	Enunciados básicos	264
6.4.2	Control de secuencia estructurado	270
6.4.3	Programas primos	279
6.5	Lecturas adicionales sugeridas	284
6.6	Problemas	284
<b>7</b>	<b>Control de subprogramas</b>	<b>286</b>
7.1	Control de secuencia de subprogramas	286
7.1.1	Subprogramas simples de llamada-regreso	288
7.1.2	Subprogramas recursivos	292
7.2	Atributos del control de datos	294
7.2.1	Nombres y ambientes de referencia	295
7.2.2	Alcance estático y dinámico	300
7.2.3	Estructura de bloques	303
7.2.4	Datos locales y ambientes locales de referencia	305
7.3	Datos compartidos en subprogramas	311
7.3.1	Parámetros y transmisión de parámetros	312
7.3.2	Ambientes comunes explícitos	330
7.3.3	Alcance dinámico	333

7.3.4 Alcance estático y estructura de bloques	337
7.4 Lecturas adicionales sugeridas	344
7.5 Problemas	345
<b>8 Abstracción II: Herencia</b>	<b>350</b>
8.1 Una nueva visita a los tipos de datos abstractos	351
8.2 Herencia	358
8.2.1 Clases derivadas	359
8.2.2 Métodos	362
8.2.3 Clases abstractas	364
8.2.4 Objetos y mensajes	366
8.2.5 Conceptos de abstracción	370
8.3 Polimorfismo	372
8.4 Lecturas adicionales sugeridas	373
8.5 Problemas	374
<b>9 Avances en el diseño de lenguajes</b>	<b>375</b>
9.1 Variaciones sobre el control de subprogramas	377
9.1.1 Excepciones y manejadores de excepciones	377
9.1.2 Corrutinas	382
9.1.3 Suprogramas planificados	383
9.1.4 Ejecución no secuencial	385
9.2 Programación en paralelo	385
9.2.1 Ejecución concurrente	387
9.2.2 Mandatos protegidos	388
9.2.3 Tareas	391
9.2.4 Sincronización de tareas	393
9.3 Propiedades formales de los lenguajes	404
9.3.1 Jerarquía de Chomsky	405
9.3.2 Irresolubilidad	408
9.3.3 Complejidad de algoritmos	413
9.4 Semántica de lenguajes	416
9.4.1 Semántica denotativa	416
9.4.2 Verificación de programas	423
9.4.3 Tipos de datos algebraicos	428
9.4.4 Resolución	431
9.5 Desarrollos del equipo (hardware)	433
9.5.1 Diseño de procesadores	433

9.5.2	Diseño de sistemas	436
9.6	Arquitectura de software	438
9.6.1	Datos persistentes y sistemas de transacción	438
9.6.2	Redes y cómputo de cliente/servidor	440
9.6.3	Edición electrónica	441
9.6.4	Tendencias de los lenguajes de programación	444
9.7	Lecturas adicionales sugeridas	444
9.8	Problemas	445
<b>Parte II. Paradigmas y lenguajes</b>		<b>449</b>
<b>10</b>	<b>Lenguajes simples de procedimientos</b>	<b>451</b>
10.1	FORTRAN	451
10.1.1	Historia	452
10.1.2	Hola Mundo	452
10.1.3	Breve perspectiva del lenguaje	453
10.1.4	Objetos de datos	457
10.1.5	Control de secuencia	462
10.1.6	Subprogramas y gestión de almacenamiento	468
10.1.7	Abstracción y encapsulamiento	470
10.1.8	Evaluación del lenguaje	471
10.2	C	471
10.2.1	Historia	472
10.2.2	Hola Mundo	472
10.2.3	Breve perspectiva del lenguaje	472
10.2.4	Objetos de datos	477
10.2.5	Control de secuencia	482
10.2.6	Subprogramas y gestión de almacenamiento	485
10.2.7	Abstracción y encapsulamiento	489
10.2.8	Evaluación del lenguaje	489
10.3	Lecturas adicionales sugeridas	490
10.4	Problemas	490
<b>11</b>	<b>Lenguajes de procedimiento estructurados en bloques</b>	<b>492</b>
11.1	Pascal	492
11.1.1	Historia	493
11.1.2	Hola Mundo	494
11.1.3	Breve perspectiva del lenguaje	494

11.1.4	Objetos de datos	498
11.1.5	Control de secuencia	505
11.1.6	Subprogramas y gestión de almacenamiento	509
11.1.7	Abstracción y encapsulamiento	516
11.1.8	Evaluación del lenguaje	516
11.2	Lecturas adicionales sugeridas	518
11.3	Problemas	518
<b>12</b>	<b>Lenguajes con base en objetos</b>	<b>520</b>
12.1	Ada	520
12.1.1	Historia	520
12.1.2	Hola Mundo	522
12.1.3	Breve perspectiva del lenguaje	522
12.1.4	Objetos de datos	527
12.1.5	Control de secuencia	536
12.1.6	Subprogramas y gestión de almacenamiento	540
12.1.7	Abstracción y encapsulamiento	547
12.1.8	Evaluación del lenguaje	549
12.2	C++	550
12.2.1	Historia	550
12.2.2	Hola Mundo	551
12.2.3	Breve perspectiva del lenguaje	551
12.2.4	Objetos de datos	556
12.2.5	Control de secuencia	561
12.2.6	Subprogramas y gestión de almacenamiento	562
12.2.7	Abstracción y encapsulamiento	564
12.2.8	Evaluación del lenguaje	564
12.3	Smalltalk	565
12.3.1	Historia	565
12.3.2	Hola Mundo	566
12.3.3	Breve perspectiva del lenguaje	566
12.3.4	Objetos de datos	570
12.3.5	Control de secuencia	572
12.3.6	Subprogramas y gestión de almacenamiento	574
12.3.7	Abstracción y encapsulamiento	577
12.3.8	Evaluación del lenguaje	577
12.4	Lecturas adicionales sugeridas	578
12.5	Problemas	579

<b>13 Lenguajes funcionales</b>	<b>581</b>
13.1 LISP	581
13.1.1 Historia	582
13.1.2 Hola Mundo	582
13.1.3 Breve perspectiva del lenguaje	583
13.1.4 Objetos de datos	587
13.1.5 Control de secuencia	589
13.1.6 Subprogramas y gestión de almacenamiento	593
13.1.7 Abstracción y encapsulamiento	599
13.1.8 Evaluación del lenguaje	599
13.2 ML	600
13.2.1 Historia	600
13.2.2 Hola Mundo	600
13.2.3 Breve perspectiva del lenguaje	601
13.2.4 Objetos de datos	603
13.2.5 Control de secuencia	607
13.2.6 Subprogramas y gestión de almacenamiento	611
13.2.7 Abstracción y encapsulamiento	613
13.2.8 Evaluación del lenguaje	616
13.3 Lecturas adicionales sugeridas	616
13.4 Problemas	617
<b>14 Lenguajes de programación lógica</b>	<b>620</b>
14.1 Prolog	620
14.1.1 Historia	621
14.1.2 Hola Mundo	621
14.1.3 Breve perspectiva del lenguaje	622
14.1.4 Objetos de datos	625
14.1.5 Control de secuencia	626
14.1.6 Subprogramas y gestión de almacenamiento	628
14.1.7 Abstracción y encapsulamiento	629
14.1.8 Evaluación del lenguaje	630
14.2 Lecturas adicionales sugeridas	630
14.3 Problemas	630
<b>Referencias</b>	<b>632</b>
<b>Índice</b>	<b>641</b>