

# Índice

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Máquinas y programas	1
1.1.1	Máquinas programables	1
1.1.2	Concepto de cómputo	3
1.1.3	Concepto de computador	4
1.2	Programación e ingeniería de software	5
1.2.1	Programación	5
1.2.2	Objetivos de la programación	6
1.3	Lenguajes de programación	7
1.4	Compiladores e Intérpretes	9
1.5	Modelos abstractos de cómputo	11
1.5.1	Modelo funcional	12
1.5.2	Modelo de flujo de datos	13
1.5.3	Modelo de programación lógica	15
1.5.4	Modelo imperativo	16
1.6	Elementos de la programación imperativa	19
1.6.1	Procesador, entorno, acciones	19
1.6.2	Acciones primitivas. Acciones compuestas	20
1.6.3	Esquemas de acciones	21
1.7	Evolución de la programación	21
1.7.1	Evolución comparativa Hardware/Software	21
1.7.2	Necesidad de metodología y buenas prácticas	22
<b>2</b>	<b>Elementos básicos de programación</b>	<b>25</b>
2.1	Lenguaje <b>C±</b>	25
2.2	Notación BNF	26
2.3	Valores y tipos	27
2.4	Representación de valores constantes	28
2.4.1	Valores numéricos enteros	29
2.4.2	Valores numéricos reales	30

---

2.4.3	Caracteres . . . . .	31
2.4.4	Cadenas de caracteres ( <i>strings</i> ) . . . . .	31
2.5	Tipos predefinidos . . . . .	32
2.5.1	El tipo entero ( <b>int</b> ) . . . . .	33
2.5.2	El tipo real ( <b>float</b> ) . . . . .	35
2.5.3	El tipo carácter ( <b>char</b> ) . . . . .	37
2.6	Expresiones aritméticas . . . . .	39
2.7	Operaciones de escritura simples . . . . .	42
2.7.1	El procedimiento <b>printf</b> . . . . .	42
2.8	Estructura de un programa completo . . . . .	45
2.8.1	Uso de comentarios . . . . .	46
2.8.2	Descripción formal de la estructura de un programa . . . . .	46
2.9	Ejemplos de programas . . . . .	47
2.9.1	Escribir una fecha . . . . .	47
2.9.2	Suma de números consecutivos . . . . .	48
2.9.3	Área y volumen de un cilindro . . . . .	49
<b>3</b>	<b>Constantes y Variables . . . . .</b>	<b>51</b>
3.1	Identificadores . . . . .	51
3.2	El vocabulario de <b>C±</b> . . . . .	53
3.3	Constantes . . . . .	54
3.3.1	Concepto de constante . . . . .	54
3.3.2	Declaración de constantes con nombre . . . . .	55
3.4	Variables . . . . .	56
3.4.1	Concepto de variable . . . . .	56
3.4.2	Declaración de variables . . . . .	58
3.4.3	Uso de variables. Inicialización . . . . .	59
3.5	Sentencia de asignación . . . . .	61
3.5.1	Sentencias de autoincremento y autodecremento . . . . .	62
3.5.2	Compatibilidad de tipos . . . . .	62
3.6	Operaciones de lectura simple . . . . .	64
3.6.1	El procedimiento <b>scanf</b> . . . . .	65
3.6.2	Lectura interactiva . . . . .	66
3.7	Estructura de un programa con declaraciones . . . . .	68
3.8	Ejemplos de programas . . . . .	68
3.8.1	Ejemplo: Conversión a horas, minutos y segundos . . . . .	68
3.8.2	Ejemplo: Área y volumen de un cilindro . . . . .	69
3.8.3	Ejemplo: Realización de un recibo . . . . .	70
<b>4</b>	<b>Metodología de Desarrollo de Programas (I) . . . . .</b>	<b>73</b>
4.1	La programación como resolución de problemas . . . . .	73

---

4.2	Descomposición en subproblemas . . . . .	74
4.3	Desarrollo por refinamientos sucesivos . . . . .	76
4.3.1	Desarrollo de un esquema secuencial . . . . .	76
4.3.2	Ejemplo: Imprimir la silueta de una silla . . . . .	79
4.4	Aspectos de estilo . . . . .	80
4.4.1	Encolumnado . . . . .	80
4.4.2	Comentarios. Documentación del refinamiento . . . . .	81
4.4.3	Elección de nombres . . . . .	85
4.4.4	Uso de letras mayúsculas y minúsculas . . . . .	88
4.4.5	Constantes con nombre . . . . .	89
4.5	Ejemplos de programas . . . . .	90
4.5.1	Ejemplo: Imprimir la figura de un árbol de navidad . . . . .	90
4.5.2	Ejemplo: Calcular el costo de las baldosas . . . . .	92
4.5.3	Ejemplo: Calcular los días entre dos fechas . . . . .	93
<b>5</b>	<b>Estructuras Básicas de la Programación Imperativa . . . . .</b>	<b>97</b>
5.1	Programación estructurada . . . . .	97
5.1.1	Representación de la estructura de un programa . . . . .	98
5.1.2	Secuencia . . . . .	99
5.1.3	Selección . . . . .	100
5.1.4	Iteración . . . . .	100
5.1.5	Estructuras anidadas . . . . .	101
5.2	Expresiones condicionales . . . . .	101
5.3	Estructuras básicas en <b>C±</b> . . . . .	106
5.3.1	Secuencia . . . . .	106
5.3.2	Sentencia IF . . . . .	106
5.3.3	Sentencia WHILE . . . . .	109
5.3.4	Sentencia FOR . . . . .	110
5.4	Ejemplos de programas . . . . .	112
5.4.1	Ejemplo: Ordenar tres datos . . . . .	112
5.4.2	Ejemplo: Escribir un triángulo de dígitos . . . . .	114
5.4.3	Ejemplo: Elaboración de tickets y resúmenes . . . . .	116
<b>6</b>	<b>Metodología de Desarrollo de Programas (II) . . . . .</b>	<b>119</b>
6.1	Desarrollo con esquemas de selección e iteración . . . . .	119
6.1.1	Esquema de selección . . . . .	120
6.1.2	Esquema de iteración . . . . .	121
6.2	Ejemplos de desarrollo con esquemas . . . . .	123
6.2.1	Ejemplo: Imprimir el contorno de un triángulo . . . . .	123
6.2.2	Ejemplo: Imprimir el triángulo de Floyd . . . . .	129
6.3	Verificación de programas . . . . .	135

6.3.1	Notación lógico-matemática . . . . .	136
6.3.2	Corrección parcial y total . . . . .	136
6.3.3	Razonamiento sobre sentencias de asignación . . . . .	137
6.3.4	Razonamiento sobre el esquema de selección . . . . .	138
6.3.5	Razonamiento sobre el esquema de iteración: invariante, terminación. . . . .	140
6.4	Eficiencia de programas. Complejidad . . . . .	142
6.4.1	Medidas de eficiencia . . . . .	142
6.4.2	Análisis de programas . . . . .	143
6.4.3	Crecimiento asintótico . . . . .	145
<b>Ejercicios sin resolver - I . . . . .</b>		<b>147</b>
<b>7</b>	<b>Funciones y Procedimientos . . . . .</b>	<b>151</b>
7.1	Concepto de subprograma . . . . .	151
7.2	Funciones . . . . .	154
7.2.1	Definición de funciones . . . . .	154
7.2.2	Uso de funciones . . . . .	156
7.2.3	Funciones predefinidas . . . . .	157
7.2.4	Funciones estándar . . . . .	158
7.3	Procedimientos . . . . .	159
7.3.1	Definición de procedimientos . . . . .	160
7.3.2	Uso de procedimientos . . . . .	161
7.3.3	Procedimientos estándar . . . . .	162
7.4	Paso de argumentos . . . . .	162
7.4.1	Paso de argumentos por valor . . . . .	162
7.4.2	Paso de argumentos por referencia . . . . .	163
7.5	Visibilidad. Estructura de bloques . . . . .	165
7.6	Recursividad de subprogramas . . . . .	167
7.7	Problemas en el uso de subprogramas . . . . .	168
7.7.1	Uso de variables globales. Efectos secundarios . . . . .	169
7.7.2	Redefinición de elementos . . . . .	171
7.7.3	Doble referencia . . . . .	173
7.8	Ejemplos de programas . . . . .	174
7.8.1	Ejemplo: Raíces de una ecuación de segundo grado . . . . .	174
7.8.2	Ejemplo: Ordenar tres valores . . . . .	176
7.8.3	Ejemplo: Perímetro de un triángulo . . . . .	178
<b>8</b>	<b>Metodología de Desarrollo de Programas (III) . . . . .</b>	<b>181</b>
8.1	Operaciones abstractas . . . . .	181
8.1.1	Especificación y realización . . . . .	182

---

8.1.2	Funciones. Argumentos . . . . .	184
8.1.3	Acciones abstractas. Procedimientos . . . . .	186
8.2	Desarrollo usando abstracciones . . . . .	188
8.2.1	Desarrollo descendente . . . . .	188
8.2.2	Ejemplo: Imprimir la figura de un árbol de navidad . . .	189
8.2.3	Ejemplo: Imprimir una tabla de números primos . . . . .	196
8.2.4	Reutilización . . . . .	199
8.2.5	Ejemplo: Tabular la serie de Fibonacci . . . . .	200
8.2.6	Desarrollo para reutilización . . . . .	203
8.2.7	Desarrollo ascendente . . . . .	206
8.3	Programas robustos . . . . .	210
8.3.1	Programación a la defensiva . . . . .	210
8.3.2	Tratamiento de excepciones . . . . .	213
<b>9</b>	<b>Definición de tipos . . . . .</b>	<b>219</b>
9.1	Tipos definidos . . . . .	219
9.2	Tipo enumerado . . . . .	221
9.2.1	Definición de tipos enumerados . . . . .	222
9.2.2	Uso de tipos enumerados . . . . .	222
9.3	El tipo predefinido <code>bool</code> . . . . .	225
9.4	Tipos estructurados . . . . .	226
9.5	Tipo formación y su necesidad . . . . .	227
9.6	Tipo vector . . . . .	228
9.6.1	Declaración de vectores . . . . .	229
9.6.2	Inicialización de un vector . . . . .	231
9.6.3	Operaciones con elementos de vectores . . . . .	231
9.6.4	Operaciones globales con vectores . . . . .	232
9.6.5	Paso de argumentos de tipo vector . . . . .	233
9.7	Vector de caracteres: Cadena ( <i>string</i> ) . . . . .	235
9.8	Tipo tupla y su necesidad . . . . .	238
9.9	Tipo registro ( <i>struct</i> ) . . . . .	239
9.9.1	Definición de registros . . . . .	240
9.9.2	Variables de tipo registro y su inicialización . . . . .	241
9.9.3	Uso de registros . . . . .	241
9.10	Ejemplos de programas . . . . .	243
9.10.1	Ejemplo: Cálculo del día de la semana de una fecha . . .	243
9.10.2	Frases palíndromas . . . . .	248
9.10.3	Ejemplo: Cálculos con fracciones . . . . .	250
<b>10</b>	<b>Ampliación de estructuras de control . . . . .</b>	<b>255</b>
10.1	Estructuras complementarias de iteración . . . . .	255

10.1.1 Repetición: Sentencia DO . . . . .	256
10.1.2 Sentencia CONTINUE . . . . .	258
10.2 Estructuras complementarias de selección . . . . .	258
10.2.1 Sentencia SWITCH . . . . .	259
10.3 Equivalencia entre estructuras . . . . .	263
10.3.1 Selección por casos . . . . .	263
10.3.2 Bucle con contador . . . . .	264
10.3.3 Repetición . . . . .	265
10.4 Ejemplos de programas . . . . .	265
10.4.1 Ejemplo: Imprimir tickets de comedor . . . . .	265
10.4.2 Ejemplo: Gestión de tarjetas de embarque . . . . .	268
10.4.3 Ejemplo: Calculadora . . . . .	273
<b>Ejercicios sin resolver - II . . . . .</b>	<b>281</b>
<b>11 Estructuras de datos . . . . .</b>	<b>285</b>
11.1 Argumentos de tipo vector abierto . . . . .	285
11.1.1 Ejemplo: Contar letras y dígitos . . . . .	287
11.2 Formaciones anidadas. Matrices . . . . .	288
11.2.1 Declaración de matrices y uso de sus elementos . . . . .	289
11.2.2 Operaciones con matrices . . . . .	290
11.2.3 Ejemplo: Contrastar una imagen . . . . .	292
11.3 El esquema unión . . . . .	295
11.3.1 El tipo <code>union</code> . . . . .	296
11.3.2 Registros con variantes . . . . .	298
11.4 Esquemas de datos y esquemas de acciones . . . . .	299
11.5 Estructuras combinadas . . . . .	300
11.5.1 Formas de combinación . . . . .	301
11.5.2 Tablas . . . . .	303
11.5.3 Ejemplo: Gestión de tarjetas de embarque . . . . .	305
<b>12 Esquemas típicos de operación con formaciones . . . . .</b>	<b>315</b>
12.1 Esquema de recorrido . . . . .	316
12.1.1 Recorrido de matrices . . . . .	318
12.1.2 Recorrido no lineal . . . . .	319
12.2 Búsqueda secuencial . . . . .	321
12.3 Inserción . . . . .	322
12.4 Ordenación por inserción directa . . . . .	324
12.5 Búsqueda por dicotomía . . . . .	326
12.6 Simplificación de las condiciones de contorno . . . . .	328
12.6.1 Técnica del centinela . . . . .	328

---

12.6.2	Matrices orladas . . . . .	330
12.7	Ejemplos de programas . . . . .	333
12.7.1	Ejemplo: Sopa de letras . . . . .	333
12.7.2	Ejemplo: Imprimir fechas en orden . . . . .	338
12.7.3	Ejemplo: Recortar una imagen . . . . .	345
<b>13</b>	<b>Punteros y variables dinámicas . . . . .</b>	<b>351</b>
13.1	Estructuras de datos no acotadas . . . . .	351
13.2	La estructura secuencia . . . . .	352
13.3	Variables dinámicas . . . . .	354
13.3.1	Punteros . . . . .	354
13.3.2	Uso de variables dinámicas . . . . .	356
13.4	Realización de secuencias mediante punteros . . . . .	358
13.4.1	Operaciones con secuencias enlazadas . . . . .	360
13.4.2	Ejemplo: Leer números y escribirlos en orden . . . . .	363
13.5	Punteros y paso de argumentos . . . . .	366
13.5.1	Paso de punteros como argumentos . . . . .	366
13.5.2	Paso de argumentos mediante punteros . . . . .	367
13.5.3	Ejemplo: Leer números y escribirlos en orden . . . . .	370
13.6	Punteros y vectores en C y C++ . . . . .	372
13.6.1	Nombres de vectores como punteros . . . . .	373
13.6.2	Paso de vectores como punteros . . . . .	374
13.6.3	Matrices y vectores de punteros . . . . .	375
<b>14</b>	<b>Tipos abstractos de datos . . . . .</b>	<b>377</b>
14.1	Concepto de tipo abstracto de datos (TAD) . . . . .	377
14.2	Realización de tipos abstractos en <b>C±</b> . . . . .	379
14.2.1	Definición de tipos abstractos como tipos registro ( <b>struct</b> )	379
14.2.2	Ocultación . . . . .	385
14.2.3	Ejemplo: Imprimir fechas en orden . . . . .	386
14.3	Metodología basada en abstracciones . . . . .	392
14.3.1	Desarrollo por refinamiento basado en abstracciones . . .	392
14.4	Ejemplo: Dibujar una <i>Curva-C</i> . . . . .	395
<b>15</b>	<b>Módulos . . . . .</b>	<b>401</b>
15.1	Concepto de módulo . . . . .	401
15.1.1	Especificación y realización . . . . .	402
15.1.2	Compilación separada . . . . .	403
15.1.3	Descomposición modular . . . . .	404
15.2	Módulos en <b>C±</b> . . . . .	405
15.2.1	Proceso de compilación simple . . . . .	406

15.2.2	Módulo principal . . . . .	407
15.2.3	Módulos no principales . . . . .	407
15.2.4	Uso de módulos . . . . .	410
15.2.5	Declaración y definición de elementos públicos . . . . .	411
15.2.6	Conflicto de nombres en el ámbito global . . . . .	412
15.2.7	Unidades de compilación en <b>C±</b> . . . . .	414
15.2.8	Compilación de programas modulares. Proyectos . . . . .	415
15.3	Desarrollo modular basado en abstracciones . . . . .	416
15.3.1	Implementación de abstracciones como módulos . . . . .	417
15.3.2	Dependencias entre ficheros. Directivas . . . . .	422
15.3.3	Datos encapsulados . . . . .	423
15.3.4	Reutilización de módulos . . . . .	429
<b>Ejercicios sin resolver - III . . . . .</b>		<b>433</b>
<b>A</b>	<b>Sintaxis de C± . . . . .</b>	<b>435</b>
A.1	Unidad de compilación . . . . .	436
A.2	Directivas de programa . . . . .	436
A.3	Declaraciones globales . . . . .	436
A.4	Declaraciones de interfaz . . . . .	437
A.5	Constantes . . . . .	437
A.6	Tipos . . . . .	437
A.7	Variables . . . . .	438
A.8	Subprogramas . . . . .	438
A.9	Bloque de código . . . . .	439
A.10	Declaraciones de bloque . . . . .	439
A.11	Sentencias ejecutables . . . . .	439
A.12	Expresiones . . . . .	440
A.13	Elementos básicos . . . . .	441
A.14	Índice de reglas BNF . . . . .	442
<b>B</b>	<b>Manual de Estilo . . . . .</b>	<b>445</b>
B.1	Aspectos generales . . . . .	446
B.1.1	Sintaxis . . . . .	446
B.1.2	Encolumnado . . . . .	446
B.1.3	Comentarios . . . . .	446
B.1.4	Identificadores . . . . .	447
B.2	Declaraciones . . . . .	447
B.2.1	Constantes . . . . .	448
B.2.2	Tipos de datos . . . . .	448
B.2.3	Variables . . . . .	450



---

B.2.4 Subprogramas . . . . .	450
B.2.5 Tipos abstractos de datos . . . . .	451
B.3 Sentencias . . . . .	452
B.4 Expresiones . . . . .	454
B.5 Punteros . . . . .	455
B.6 Módulos . . . . .	455
<b>C Notación lógico-matemática . . . . .</b>	<b>457</b>
C.1 Operadores numéricos . . . . .	457
C.2 Operadores de comparación . . . . .	458
C.3 Operadores lógicos . . . . .	458
C.4 Colecciones . . . . .	458
C.5 Cuantificadores . . . . .	459
C.6 Expresiones condicionales . . . . .	459
<b>Bibliografía . . . . .</b>	<b>461</b>
<b>Índice analítico . . . . .</b>	<b>463</b>