



<b>Visão Geral da Obra .....</b>	<b>v</b>
<b>Conteúdo Resumido .....</b>	<b>vii</b>
<b>Conteúdo Completo .....</b>	<b>ix</b>
<b>Lista de Figuras .....</b>	<b>xxv</b>
<b>Lista de Tabelas .....</b>	<b>xxxI</b>
<b>Lista de Algoritmos .....</b>	<b>xxxiii</b>
<b>Lista de Lemas, Teoremas e Corolários .....</b>	<b>xxxv</b>
<b>Prefácio .....</b>	<b>xxxvii</b>
Objetivos e Público-alvo .....	xxxvii
Organização do Livro .....	xxxvii
Volume 1 .....	xxxvii
Volume 2 .....	xxxix
Exemplos de Programação .....	xl
Exercícios .....	xl
Organizadores Prévios .....	xli
Material Complementar .....	xli
Recursos Utilizados .....	xli
Agradecimentos .....	xli

## PARTE 1: REVISÃO DA LINGUAGEM C

1 Elementos Básicos da Linguagem C..... 45

1.1	A Linguagem C Padrão	46
1.2	Identificadores, Tipos Primitivos e Constantes	46
1.2.1	Identificadores	46
1.2.2	Tipos de Dados Primitivos	46
1.2.3	Constantes	47
1.3	Operadores Básicos	48
1.3.1	Propriedades de Operadores	48
1.3.2	Expressões	49
1.3.3	Operadores Aritméticos	49
1.3.4	Operadores Relacionais	49
1.3.5	Operadores Lógicos	49
1.4	Definições de Variáveis	50
1.5	Operadores de Atribuição	50
1.6	Conversões de Tipos	51
1.6.1	Conversões Implícitas	51
1.6.2	Conversão Explícita	52
1.7	Incremento e Decremento	52
1.8	Comentários	53
1.9	A Biblioteca Padrão de C	53
1.10	Escrita de Dados na Tela	54
1.11	Leitura de Dados via Teclado 1: Frágil	55
1.11.1	Função <i>getchar()</i>	55
1.11.2	Função <i>scanf()</i>	56
1.12	Constantes Simbólicas	57
1.13	Enumerações	57
1.14	Instruções	58
1.14.1	Tipos de Instruções	58
1.14.2	Sequências de Instruções	58
1.14.3	Instruções Vazias	58
1.15	Estruturas de Controle	59
1.15.1	Laços de Repetição	59
	<i>while</i>	59
	<i>do-while</i>	60
	<i>for</i>	60
	Laços de Repetição Aninhados	62
	Laços de Repetição Infinitos	62
1.15.2	Desvios Condicionais	62
	<i>if-else</i>	62
	<i>switch-case</i>	63
1.15.3	Desvios Incondicionais	65
	<i>break</i>	65
	<i>continue</i>	65

	<i>goto</i>	65
	Uso de <i>break</i> e <i>continue</i> em Laços Aninhados	66
1.16	Outros Operadores	66
1.16.1	Operador Condicional	66
1.16.2	Operador Vírgula	67
1.16.3	O Operador <i>sizeof</i> e o Tipo <i>size_t</i>	67
1.17	Programas Monoarquivo	68
1.17.1	Estrutura de um Programa Simples em C	68
1.17.2	Como Criar um Programa-fonte	68
1.17.3	Criando um Programa Executável	68
1.18	Endereços e Ponteiros	69
1.18.1	Endereços	69
1.18.2	Ponteiros	69
1.18.3	Indireção de Ponteiros	70
1.18.4	Ponteiro Nulo	71
1.19	Exercícios de Revisão	71
	A Linguagem C Padrão (Seção 1.1)	71
	Identificadores, Tipos Primitivos e Constantes (Seção 1.2)	71
	Operadores Básicos (Seção 1.3)	72
	Definições de Variáveis (Seção 1.4)	72
	Operadores de Atribuição (Seção 1.5)	72
	Conversões de Tipos (Seção 1.6)	73
	Incremento e Decremento (Seção 1.7)	73
	Comentários (Seção 1.8)	73
	A Biblioteca Padrão de C (Seção 1.9)	73
	Escrita de Dados na Tela (Seção 1.10)	74
	Leitura de Dados via Teclado 1: Frágil (Seção 1.11)	74
	Constantes Simbólicas (Seção 1.12)	75
	Enumerações (Seção 1.13)	75
	Instruções (Seção 1.14)	75
	Estruturas de Controle (Seção 1.15)	75
	Outros Operadores (Seção 1.16)	77
	Programas Monoarquivo (Seção 1.17)	78
	Endereços e Ponteiros (Seção 1.18)	78
1.20	Exercícios de Programação	78

2 Funções e Programas Multiarquivo ..... 81

2.1	Funções	82
2.1.1	Definições de Funções	82
2.1.2	Cabeçalho	82
2.1.3	Corpo de Função	83
2.1.4	Retorno de Função	83
2.1.5	Chamadas de Funções	84
2.1.6	Modos de Parâmetros	84

2.1.7	Passagem de Parâmetros	85
2.1.8	Simulando Passagem por Referência em C	85
2.1.9	Alusões e Protótipos de Funções	87
2.2	Leitura de Dados via Teclado 2: Robusta	87
2.2.1	Entendendo Leitura de Dados via Teclado	87
2.2.2	Esvaziamento do Buffer de Entrada	88
2.2.3	Uso de Laços de Repetição em Leitura de Dados	89
2.3	Duração de Variáveis	90
2.4	Escopo	91
2.5	Diretivas de Pré-processamento	91
2.5.1	Macros	92
2.5.2	Compilação Condicional	94
2.5.3	Inclusão de Arquivos	95
2.6	Programas Multiarquivo	96
2.6.1	Variáveis Globais	96
2.6.2	Funções de Arquivo e Globais	97
2.6.3	Módulos	97
2.6.4	Como Construir Programas Multiarquivo	98
	Utilizando um Ambiente Integrado de Desenvolvimento	98
	Utilizando Editor de Programas e Compilador	99
	Make e Arquivos Makefiles	100
2.7	Exemplos de Programação	103
2.7.1	Um Módulo para Leitura de Dados Resiliente	103
2.7.2	Comparando Números Reais	107
2.7.3	Série de Taylor para Cálculo de Seno	107
2.7.4	Raiz Quadrada Usando o Método de Newton e Raphson	109
2.8	Exercícios de Revisão	110
	Funções (Seção 2.1)	110
	Leitura de Dados via Teclado 2: Robusta (Seção 2.2)	112
	Duração de Variáveis (Seção 2.3)	112
	Escopo (Seção 2.4)	113
	Diretivas de Pré-processamento (Seção 2.5)	114
	Programas Multiarquivo (Seção 2.6)	114
	Exemplos de Programação (Seção 2.7)	115
2.9	Exercícios de Programação	115
3	Tipos de Dados Estruturados .....	119
3.1	Arrays	120
3.1.1	Definições de Arrays	120
3.1.2	Acesso a Elementos de um Array	120
3.1.3	Iniciações de Arrays	121
3.1.4	Tamanho de Array	121
3.2	Aritmética de Ponteiros	122
3.3	Relações entre Ponteiros e Arrays	123

3.4	O Qualificador const	124
3.5	Uso de Arrays com Funções	125
3.5.1	Declarando Arrays como Parâmetros Formais	125
3.5.2	Arrays como Parâmetros Reais	126
3.5.3	Retorno de Arrays e Zumbis	126
3.5.4	Qualificação de Parâmetros com const	127
3.6	Arrays Multidimensionais	127
3.6.1	Iniciações de Arrays Multidimensionais	127
3.6.2	Acesso a Elementos de Arrays Multidimensionais	128
3.6.3	Arrays Multidimensionais como Parâmetros de Funções	129
3.7	Strings e Caracteres	129
3.7.1	Strings	129
3.7.2	Funções de Biblioteca para Processamento de Strings	130
3.7.3	Conversão de Strings Numéricos em Números	131
3.7.4	Classificação de Caracteres: Funções isX()	132
3.7.5	Transformação de Caracteres: tolower() e toupper()	132
3.8	Função main() com Parâmetros	132
3.9	Tipos Definidos pelo Programador	133
3.10	Estruturas e Uniões	133
3.10.1	Definições de Estruturas	134
3.10.2	Estrutura com Autorreferência	134
3.10.3	Iniciações de Estruturas	135
3.10.4	Atribuições entre Estruturas	135
3.10.5	Acesso a Campos de Estruturas	135
3.10.6	Aninho de Estruturas	136
3.10.7	Estruturas como Parâmetros de Funções	137
3.10.8	Funções com Retorno de Estruturas	137
3.10.9	Uniões	138
3.10.10	Registros Variantes	138
3.11	Operadores de Acesso e Definidores de Tipos	139
3.12	Exemplos de Programação	141
3.12.1	Incrementando o Módulo de Leitura Resiliente	141
3.12.2	Ordenação de Arrays pelo Método da Bolha	143
3.12.3	Validando Datas	144
3.12.4	Operações com Vetores Reais	146
3.12.5	Coordenadas Retangulares e Polares	148
3.13	Exercícios de Revisão	151
	Arrays (Seção 3.1)	151
	Aritmética de Ponteiros (Seção 3.2)	152
	Relações entre Ponteiros e Arrays (Seção 3.3)	152
	O Qualificador const (Seção 3.4)	153
	Uso de Arrays com Funções (Seção 3.5)	153
	Arrays Multidimensionais (Seção 3.6)	155
	Strings e Caracteres (Seção 3.7)	155

Função main() com Parâmetros (Seção 3.8)	156
Tipos Definidos pelo Programador (Seção 3.9)	157
Estruturas e Uniãoes (Seção 3.10)	157
Operadores de Acesso e Definidores de Tipos (Seção 3.11)	157
Exemplos de Programação (Seção 3.12)	158
3.14 Exercícios de Programação	158
<b>4 Recursão e Retrocesso</b>	<b>163</b>
4.1 Funções Recursivas	164
4.2 Cadeias Recursivas	166
4.3 Pilha de Execução e Registros de Ativação	166
4.4 Recursão de Cauda	170
4.5 Retrocesso (Backtracking)	173
4.5.1 O Problema das N Rainhas	174
Entendendo o Problema	175
Implementação da Solução	178
4.5.2 Outros Problemas Propícios ao Uso de Retrocesso	190
4.6 Como Pensar Recursivamente	190
4.7 Quando Usar (e Não Usar) Recursão	192
4.8 Exemplos de Programação	195
4.8.1 O (Cansativo) Problema das Torres de Hanói	195
4.8.2 Fibonacci + Recursão = Ineficiência 1	198
4.8.3 Calculando o Comprimento de um String Recursivamente	200
4.8.4 Removendo Vogais de um String Recursivamente	200
4.8.5 Exponenciação por Quadratura 1	201
4.8.6 Invertendo Entradas 1	202
4.8.7 Exibindo-se em Frente e Verso	202
4.8.8 Resolvendo Sudoku	204
4.9 Exercícios de Revisão	207
Funções Recursivas (Seção 4.1)	207
Cadeias Recursivas (Seção 4.2)	210
Pilha de Execução e Registros de Ativação (Seção 4.3)	211
Recursão de Cauda (Seção 4.4)	211
Retrocesso (Backtracking) (Seção 4.5)	211
Como Pensar Recursivamente (Seção 4.6)	211
Quando Usar (e Não Usar) Recursão (Seção 4.7)	212
Exemplos de Programação (Seção 4.8)	212
4.10 Exercícios de Programação	212

PARTE 2: FUNDAMENTOS

<b>5 Conceitos Básicos de Estruturas de Dados</b>	<b>217</b>
5.1 Definições Fundamentais	218
5.2 Tipos Abstratos e Estruturas de Dados	219

5.3 Algoritmos	220
5.3.1 Abordagem de Refinamentos Sucessivos	221
5.3.2 Linguagem Algorítmica (Pseudolinguagem)	221
5.3.3 Como Construir Algoritmos	221
5.3.4 Paradigmas Algorítmicos	222
Algoritmos de Divisão e Conquista	222
Algoritmos de Retrocesso	222
Algoritmos Vorazes	223
Algoritmos de Força Bruta	223
5.4 Tipos de Dados Transparentes e TADs em C	223
5.5 Exemplos de Programação	225
5.5.1 Número Complexo como um Tipo de Dado Transparente	225
Interface	225
Implementação	226
5.5.2 Número Complexos como um TAD	227
Interface	227
Implementação	228
5.6 Exercícios de Revisão	229
Definições Fundamentais (Seção 5.1)	229
Tipos Abstratos e Estruturas de Dados (Seção 5.2)	230
Algoritmos (Seção 5.3)	230
Tipos de Dados Transparentes e TADs em C (Seção 5.4)	231
Exemplos de Programação (Seção 5.5)	231
5.7 Exercícios de Programação	231
<b>6 Análise de Algoritmos</b>	<b>235</b>
6.1 Complexidade de Algoritmos	236
6.2 Análise Assintótica	237
6.3 Notações Ó, Ômega e Teta	238
6.4 Casos Melhor, Pior e Mediano	244
6.5 Funções Comuns em Análise de Algoritmos	246
6.6 Resultados Teóricos Importantes	249
6.7 Regras Práticas para Análise Temporal de Algoritmos	250
6.7.1 Análise Temporal de Instruções Simples	251
6.7.2 Análise Temporal de Sequências de Instruções	251
6.7.3 Análise Temporal de Estruturas de Controle	251
Desvios Condicionais	251
Desvios Incondicionais e Instruções de Retorno	252
Laços de Repetição	252
6.7.4 Análise Temporal de Chamadas de Funções	252
6.8 Análise de Custo Espacial de Algoritmos	253
6.9 Algoritmos Recursivos e Relações de Recorrência	254
6.10 Árvores de Recursão	255
6.10.1 Conceitos	255

6.10.2	Exemplos	257
	Equação de Recorrência 1: $T(n) = 2 \cdot T(n/2) + n$	257
	Equação de Recorrência 2: $T(n) = T(n/2) + n$	258
	Equação de Recorrência 3: $4 \cdot T(n) = T(n/2) + n$	259
6.10.3	Análise	260
6.11	Exemplos de Programação	261
6.11.1	Um Algoritmo com Custo Temporal $\theta(1)$	261
6.11.2	Um Algoritmo com Custo Temporal $\theta(n)$	261
6.11.3	Um Algoritmo com Custo Temporal $\theta(n^2)$	261
6.11.4	Um Algoritmo com Custo Temporal $\theta(\log n)$	262
6.11.5	Um Algoritmo com Custo Temporal $\theta(n \log n)$	263
6.11.6	Quando Será o Fim do Mundo?	264
6.11.7	Fibonacci + Recursão = Ineficiência 2	266
6.11.8	Exponenciação por Quadratura 2	268
6.12	Exercícios de Revisão	268
	Complexidade de Algoritmos (Seção 6.1)	268
	Análise Assintótica (Seção 6.2)	268
	Notações Ó, Ômega e Teta (Seção 6.3)	269
	Casos Melhor, Pior e Mediano (Seção 6.4)	270
	Funções Comuns em Análise de Algoritmos (Seção 6.5)	270
	Resultados Teóricos Importantes (Seção 6.6)	271
	Regras Práticas para Análise Temporal de Algoritmos (Seção 6.7)	272
	Análise de Custo Espacial de Algoritmo (Seção 6.8)	275
	Algoritmos Recursivos e Relações de Recorrência (Seção 6.9)	275
	Árvores de Recursão (Seção 6.10)	275
	Exemplos de Programação (Seção 6.11)	276
6.13	Exercícios de Programação	276

PARTE 3: ESTRUTURAS DE DADOS BÁSICAS

7

Listas Indexadas

279

7.1	Listas sem Ordenação	280
7.1.1	Abstração	280
7.1.2	Implementação Usando Arrays Estáticos	281
	Definições de Tipos	282
	Iniciação	282
	Comprimento	282
	Acesso	282
	Inserção	283
	Acréscimo	284
	Remoção	285
	Checagem de Lista Vazia	286
7.1.3	Busca Sequencial	286
7.2	Listas Ordenadas	288

7.2.1	Abstração	288
7.2.2	Implementação Usando Arrays Estáticos	288
7.2.3	Busca Binária	289
7.3	Análise de Operações sobre Listas Indexadas	292
7.4	Tratamento de Exceções	293
7.5	Aplicações de Listas	296
7.5.1	Representação de Polinômios	296
7.5.2	Representação de Matrizes Esparsas	298
7.6	Exemplos de Programação	301
7.6.1	Gerenciando uma Lista de Compras com Sensibilidade	301
7.6.2	Lista Ordenada Armazenada em Arquivo	307
7.6.3	Removendo Duplicatas de uma Lista	315
7.7	Exercícios de Revisão	316
	Listas sem Ordenação (Seção 7.1)	316
	Listas Ordenadas (Seção 7.2)	317
	Análise de Operações sobre Listas Indexadas (Seção 7.3)	317
	Tratamento de Exceções (Seção 7.4)	317
	Aplicações de Listas (Seção 7.5)	317
	Exemplos de Programação (Seção 7.6)	318
7.8	Exercícios de Programação	318

8

Pilhas e Filas

321

8.1	Pilhas	322
8.1.1	Conceito	322
8.1.2	Implementação	324
8.1.3	Análise	325
8.2	Filas Lineares	325
8.2.1	Conceito	325
8.2.2	Implementação	326
8.2.3	Análise	329
8.3	Filas Circulares	329
8.4	Transformando Recursão em Iteração Usando Pilha	333
8.5	Exemplos de Programação	338
8.5.1	Invertendo Entradas 2	338
8.5.2	Casamento de Parênteses, Colchetes e Chaves	339
8.5.3	Testando Palíndromos	342
8.5.4	Fazendo, Desfazendo e Refazendo	343
8.5.5	Simulação de Fila de Banco	352
8.6	Exercícios de Revisão	355
	Pilhas (Seção 8.1)	355
	Filas Lineares (Seção 8.2)	357
	Filas Circulares (Seção 8.3)	358
	Transformando Recursão em Iteração Usando Pilha (Seção 8.4)	358
	Exemplos de Programação (Seção 8.5)	358



8.7	Exercícios de Programação	359
9	Alocação Dinâmica de Memória	361
9.1	Motivação e Justificativas	362
9.2	Funções de Alocação e Liberação Dinâmica de Memória	363
9.2.1	malloc()	363
9.2.2	calloc()	363
9.2.3	free()	364
9.2.4	realloc()	365
9.3	Ponteiros Genéricos e o Tipo void *	366
9.4	A Partição de Memória Heap	368
9.5	Testando Alocação Dinâmica de Memória	369
9.6	Listas Indexadas Dinâmicas	369
9.6.1	Lista sem Ordenação	370
	Definições de Tipos e Constantes	370
	Criação	370
	Destruição	371
	Inserção	371
	Remoção	372
	Acréscimo de Elemento	373
9.6.2	Lista Ordenada	373
	Definições de Tipos e Constantes	373
	Criação	374
	Inserção em Ordem	374
	Remoção	375
9.7	Pilhas e Filas Implementadas com Arrays Dinâmicos	376
9.7.1	TAD Pilha 1	376
	Arquivo de Cabeçalho	376
	Arquivo de Implementação	376
9.7.2	TAD Fila 1	378
	Arquivo de Cabeçalho	378
	Arquivo de Implementação	378
9.8	Análise de Implementações com Arrays Dinâmicos	380
9.9	Exemplos de Programação	381
9.9.1	Lendo Linhas (Praticamente) Ilimitadas	381
9.9.2	A Urupema de Eratóstenes	387
9.9.3	Concatenação Múltipla de Strings	390
9.10	Exercícios de Revisão	392
	Motivação e Justificativas (Seção 9.1)	392
	Funções de Alocação e Liberação Dinâmica de Memória (Seção 9.2)	392
	Ponteiros Genéricos e o Tipo void * (Seção 9.3)	393
	A Partição de Memória Heap (Seção 9.4)	393
	Testando Alocação Dinâmica de Memória (Seção 9.5)	393
	Listas Indexadas Dinâmicas (Seção 9.6)	394

	Pilhas e Filas Implementadas com Arrays Dinâmicos (Seção 9.7)	394
	Análise de Implementações com Arrays Dinâmicos (Seção 9.8)	394
	Exemplos de Programação (Seção 9.9)	395
9.11	Exercícios de Programação	395
10	Listas Encadeadas	399
10.1	Deficiências de Arrays Dinâmicos	400
10.2	Lista Simplesmente Encadeada sem Ordenação	401
10.2.1	Abstração	401
10.2.2	Implementação	401
	Definições de Tipos	401
	Iniciação	403
	Checagem de Lista Vazia	403
	Comprimento	403
	Inserção no Início	403
	Remoção	405
	Busca	408
	Destruição	409
	Acesso Sequencial	410
10.3	Lista Simplesmente Encadeada com Ordenação	411
10.3.1	Abstração	411
10.3.2	Implementação	412
	Inserção em Ordem	412
	Busca	415
10.4	Outros Tipos de Listas Encadeadas	415
10.4.1	Lista Duplamente Encadeada Linear	416
	Inserção	416
	Remoção	417
	Acesso Sequencial Invertido	418
10.4.2	Lista Simplesmente Encadeada Circular	419
	Inserção	419
	Remoção	421
	Busca	422
	Destruição	422
	Acesso Sequencial	422
10.4.3	Lista Duplamente Encadeada Circular	422
	Inserção	422
	Remoção	424
	Busca	425
	Destruição	425
	Acesso Sequencial	425
	Acesso Sequencial Invertido	425
10.4.4	Lista Encadeada com Cabeça	426
	Iniciação	426

	Comprimento	426
	Inserção	427
	Remoção	427
	Busca	427
	Destruição	427
	Acessos Sequenciais Direto e Invertido	427
<b>10.5</b>	<b>Pilhas e Filas Encadeadas</b>	<b>428</b>
10.5.1	TAD Pilha 2	428
	Arquivo de Cabeçalho	428
	Arquivo de Implementação	428
10.5.2	TAD Fila 2	430
	Arquivo de Cabeçalho	430
	Arquivo de Implementação	430
<b>10.6</b>	<b>Análise de Operações sobre Listas Encadeadas</b>	<b>432</b>
<b>10.7</b>	<b>Exemplos de Programação</b>	<b>432</b>
10.7.1	Invertendo uma Lista Simplesmente Encadeada	432
10.7.2	Números Felizes	434
10.7.3	Representação de Polinômios Usando Listas Encadeadas	436
10.7.4	A História de Josephus	442
10.7.5	Números Inteiros de Larguras Ilimitadas	445
<b>10.8</b>	<b>Exercícios de Revisão</b>	<b>455</b>
	Deficiências de Arrays Dinâmicos (Seção 10.1)	455
	Lista Simplesmente Encadeada sem Ordenação (Seção 10.2)	455
	Lista Simplesmente Encadeada com Ordenação (Seção 10.3)	457
	Outros Tipos de Listas Encadeadas (Seção 10.4)	458
	Pilhas e Filas Encadeadas (Seção 10.5)	460
	Análise de Operações sobre Listas Encadeadas (Seção 10.6)	460
	Exemplos de Programação (Seção 10.7)	460
<b>10.9</b>	<b>Exercícios de Programação</b>	<b>461</b>
<b>11</b>	<b>Estruturas de Dados e Algoritmos Genéricos</b>	<b>465</b>
<b>11.1</b>	<b>Ponteiros para Funções</b>	<b>466</b>
11.1.1	Definição de Ponteiro para Função	466
11.1.2	Atribuição de Valor a um Ponteiro para Função	467
11.1.3	Chamada de Função Mediante Ponteiro	468
11.1.4	Retorno de Ponteiro para Função	468
11.1.5	Ponteiro para Função como Parâmetro de Função	469
<b>11.2</b>	<b>Listas Generalizadas</b>	<b>470</b>
11.2.1	Conceitos	470
11.2.2	Implementação	471
11.2.3	Funções Recursivas para Listas Generalizadas	472
	Clonagem	472
	Igualdade	472
	Profundidade	473

	Exibição	474
<b>11.3</b>	<b>Estruturas de Dados Genéricas</b>	<b>474</b>
<b>11.4</b>	<b>Implementação de Pilha Genérica</b>	<b>475</b>
11.4.1	Definição de Tipo	475
11.4.2	Criação	475
11.4.3	Destruição	476
11.4.4	Verificação de Pilha Vazia	476
11.4.5	Empilhamento	476
11.4.6	Desempilhamento	477
11.4.7	Elemento do Topo	478
<b>11.5</b>	<b>Usando bsearch() e qsort()</b>	<b>478</b>
<b>11.6</b>	<b>Conversões e Avaliações de Expressões Aritméticas</b>	<b>480</b>
11.6.1	Formas de Representação de Expressões Aritméticas	480
11.6.2	Avaliação de Expressão Sufixa	482
11.6.3	Conversão de Forma Infixa para Forma Sufixa	482
<b>11.7</b>	<b>Exemplos de Programação</b>	<b>486</b>
11.7.1	Ordenação Generalizada de Listas Indexadas	486
11.7.2	Implementando Conversão de Expressões Infixas em Sufixas	487
11.7.3	Implementando Avaliação de Expressões Sufixas	492
<b>11.8</b>	<b>Exercícios de Revisão</b>	<b>495</b>
	Ponteiros para Funções (Seção 11.1)	495
	Listas Generalizadas (Seção 11.2)	495
	Estruturas de Dados Genéricas (Seção 11.3)	496
	Implementação de Pilha Genérica (Seção 11.4)	496
	Usando bsearch() e qsort() (Seção 11.5)	496
	Conversões e Avaliações de Expressões Aritméticas (Seção 11.6)	497
	Exemplos de Programação (Seção 11.7)	498
<b>11.9</b>	<b>Exercícios de Programação</b>	<b>498</b>
<b>12</b>	<b>Árvores</b>	<b>501</b>
<b>12.1</b>	<b>Conceitos Fundamentais</b>	<b>502</b>
<b>12.2</b>	<b>Árvores Binárias</b>	<b>504</b>
12.2.1	Degeneradas (ou Patológicas)	505
12.2.2	Estritamente Binárias	505
12.2.3	Perfeitas (ou Repletas)	505
12.2.4	Completas	506
<b>12.3</b>	<b>Caminhamentos em Árvores Binárias</b>	<b>508</b>
12.3.1	Caminhamento Prefixo	508
12.3.2	Caminhamento Infixo	509
12.3.3	Caminhamento Sufixo	509
12.3.4	Exemplos	509
12.3.5	Caminhamento em Largura (ou por Nível)	512
<b>12.4</b>	<b>Implementação de Árvores Binárias</b>	<b>512</b>
12.4.1	Definição de Tipo	512

12.4.2 Criação de Nós	513
12.4.3 Caminhamentos	514
12.4.4 Altura	514
12.4.5 Semelhança	515
12.4.6 Clonagem	516
12.4.7 Contagem de Nós	516
12.4.8 Destruição	516
12.5 Árvores Binárias Baseadas em Caminhamentos	516
12.6 Árvores Binárias Costuradas	520
12.7 Conversões de Árvores Ordinárias e Florestas em Árvores Binárias	523
12.8 Exemplos de Programação	526
12.8.1 Problema das Oito Moedas	526
12.8.2 Codificação de Huffman	527
12.9 Exercícios de Revisão	540
Conceitos Fundamentais (Seção 12.1)	540
Árvores Binárias (Seção 12.2)	541
Caminhamentos em Árvores Binárias (Seção 12.3)	544
Implementação de Árvores Binárias (Seção 12.4)	546
Árvores Binárias Baseadas em Caminhamentos (Seção 12.5)	548
Árvores Binárias Costuradas (Seção 12.6)	548
Conversões de Árvores Ordinárias e Florestas em Árvores Binárias (Seção 12.7)	548
Exemplos de Programação (Seção 12.8)	549
12.10 Exercícios de Programação	550

APÊNDICES

A Precedência e Associatividade de Operadores da Linguagem C	555
--	-----

B Elementos de Matemática	557
---------------------------	-----

B.1 Somatórios	557
B.2 Polinômios	558
B.3 O Método de Horner	559
B.4 Logaritmos	561
B.5 Matrizes	561
B.6 Funções Piso e Teto	562
B.7 Análise Combinatória	563
B.8 Indução Matemática	564
B.8.1 Indução Fraca	564
B.8.2 Indução Forte	565
B.9 Relações de Recorrência	566
B.9.1 Conjectura e Indução Matemática	567
B.9.2 Relações de Recorrência Homogêneas	567
B.10 Propriedades de Análise Assintótica	568

B.11 Propriedades de Árvores Binárias	571
---------------------------------------	-----

C Guia de Nomenclatura Usada em Identificadores	575
---	-----

C.1 Regras Básicas de Escrita de Identificadores	575
C.1.1 Variáveis e Parâmetros Formais	575
C.1.2 Macros e Constantes de Enumeração	576
C.1.3 Funções	576
C.1.4 Tipos	576
C.1.5 Rótulos de Estruturas	576
C.2 Glossário de Sufixos de Identificadores	576

D Respostas e Sugestões para os Exercícios de Revisão	579
---	-----

Capítulo 1 — Elementos Básicos da Linguagem C	579
Capítulo 2 — Funções e Programas Multiarquivo	584
Capítulo 3 — Tipos de Dados Estruturados	588
Capítulo 4 — Recursão e Retrocesso	592
Capítulo 5 — Conceitos Básicos de Estruturas de Dados	594
Capítulo 6 — Análise de Algoritmos	595
Capítulo 7 — Listas Indexadas	601
Capítulo 8 — Pilhas e Filas	603
Capítulo 9 — Alocação Dinâmica de Memória	606
Capítulo 10 — Listas Encadeadas	608
Capítulo 11 — Estruturas de Dados e Algoritmos Genéricos	612
Capítulo 12 — Árvores	613

REFERÊNCIAS

Bibliografia	623
--------------	-----

Índice Remissivo	625
------------------	-----



