

# A Linguagem de Programação Go

**Alan A. A. Donovan**

Google Inc.

**Brian W. Kernighan**

Princeton University

Tradução

**Lúcia A. Kinoshita**

Revisão técnica

**Luciano Ramalho (ThoughtWorks, Inc.)**



Addison  
Wesley

**Novatec**

Authorized translation from the English language edition, entitled GO PROGRAMMING LANGUAGE, THE, 1st Edition, by ALAN DONOVAN; BRIAN KERNIGHAN, published by Pearson Education, Inc, publishing as Addison-Wesley Professional, Copyright © 2017 by Alan A. A. Donovan & Brian W. Kernighan.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc. PORTUGUESE language edition published by NOVATEC EDITORA LTDA., Copyright © 2017.

Tradução autorizada da edição original em inglês, intitulada GO PROGRAMMING LANGUAGE, THE, 1st Edition, por ALAN DONOVAN; BRIAN KERNIGHAN, publicada pela Pearson Education, Inc, publicando como Addison-Wesley Professional, Copyright © 2017 por Alan A. A. Donovan & Brian W. Kernighan.

Todos os direitos reservados. Nenhuma parte deste livro pode ser reproduzida ou transmitida por qualquer forma ou meio, eletrônica ou mecânica, incluindo fotocópia, gravação ou qualquer sistema de armazenamento de informação, sem a permissão da Pearson Education, Inc. Edição em Português publicada pela NOVATEC EDITORA LTDA., Copyright © 2017.

Todos os direitos reservados e protegidos pela Lei 9610 de 19/02/1998. É proibida a reprodução desta obra, mesmo parcial, por qualquer processo, sem prévia autorização, por escrito, do autor e da Editora.

Editor: Rubens Prates  
Assistente editorial: Priscila A. Yoshimatsu  
Tradução: Lúcia A. Kinoshita  
Revisão técnica: Luciano Ramalho (ThoughtWorks, Inc.)  
Editoração eletrônica: Carolina Kuwabata  
Revisão gramatical: Smirna Cavalheiro/Solange Martins

ISBN: 978-85-7522-546-2

Histórico de impressões:

Março/2017                      Primeira edição

Novatec Editora Ltda.  
Rua Luís Antônio dos Santos 110  
02460-000 – São Paulo, SP – Brasil  
Tel.: +55 11 2959-6529  
Email: [novatec@novatec.com.br](mailto:novatec@novatec.com.br)  
Site: [www.novatec.com.br](http://www.novatec.com.br)  
Twitter: [twitter.com/novateceditora](https://twitter.com/novateceditora)  
Facebook: [facebook.com/novatec](https://facebook.com/novatec)  
LinkedIn: [linkedin.com/in/novatec](https://linkedin.com/in/novatec)

# Sumário

---

<b>Prefácio .....</b>	<b>13</b>
Origens de Go .....	14
Projeto Go .....	16
Como este livro está organizado .....	18
Onde encontrar mais informações .....	20
Agradecimentos .....	21
<b>1 Tutorial .....</b>	<b>22</b>
1.1 Hello, World .....	22
1.2 Argumentos de linha de comando .....	25
1.3 Encontrando linhas duplicadas .....	31
1.4 GIFs animados .....	37
1.5 Buscando um URL .....	40
1.6 Buscando URLs de modo concorrente .....	42
1.7 Um servidor web .....	44
1.8 Miscelâneas .....	50
<b>2 Estrutura dos programas .....</b>	<b>53</b>
2.1 Nomes .....	53
2.2 Declarações .....	55
2.3 Variáveis .....	56
2.3.1 Declarações curtas de variáveis .....	57
2.3.2 Ponteiros .....	59
2.3.3 A função new .....	62
2.3.4 Tempo de vida das variáveis .....	63
2.4 Atribuições .....	65
2.4.1 Atribuição de tupla .....	65
2.4.2 Possibilidade de atribuição .....	67
2.5 Declarações de tipos .....	68
2.6 Pacotes e arquivos .....	71
2.6.1 Importações .....	73
2.6.2 Inicialização de pacotes .....	75
2.7 Escopo .....	77

<b>3</b>	<b>Tipos de dados básicos.....</b>	<b>83</b>
3.1	Inteiros .....	83
3.2	Números de ponto flutuante .....	90
3.3	Números complexos .....	95
3.4	Booleanos.....	98
3.5	Strings.....	99
3.5.1	Strings literais .....	101
3.5.2	Unicode .....	103
3.5.3	UTF-8 .....	104
3.5.4	Strings e fatias de bytes.....	108
3.5.5	Conversões entre strings e números .....	113
3.6	Constantes .....	113
3.6.1	Gerador de constantes iota .....	115
3.6.2	Constantes sem tipo .....	117
<b>4</b>	<b>Tipos compostos.....</b>	<b>121</b>
4.1	Arrays .....	121
4.2	Fatias .....	125
4.2.1	Função append .....	130
4.2.2	Técnicas in-place para fatias .....	134
4.3	Mapas.....	136
4.4	Estruturas .....	144
4.4.1	Estruturas literais.....	148
4.4.2	Comparando estruturas.....	149
4.4.3	Inclusão de estruturas e campos anônimos.....	150
4.5	JSON .....	153
4.6	Templates de texto e HTML .....	161
<b>5</b>	<b>Funções .....</b>	<b>168</b>
5.1	Declarações de funções.....	168
5.2	Recursão .....	170
5.3	Múltiplos valores de retorno .....	175
5.4	Erros .....	178
5.4.1	Estratégias de tratamento de erros.....	180
5.4.2	Fim de arquivo (EOF) .....	184
5.5	Valores função .....	185
5.6	Funções anônimas .....	188
5.6.1	Cuidado: captura de variáveis de iteração .....	195
5.7	Funções variádicas .....	197
5.8	Chamadas de função adiadas .....	199

5.9	Pânico .....	205
5.10	Recuperação.....	209
<b>6</b>	<b>Métodos .....</b>	<b>213</b>
6.1	Declarações de métodos .....	214
6.2	Métodos cujo receptor é um ponteiro .....	216
6.2.1	Nil é um valor válido de receptor .....	219
6.3	Compondo tipos por meio de inclusão de estruturas .....	221
6.4	Valores e expressões método .....	224
6.5	Exemplo: tipo vetor de bits.....	226
6.6	Encapsulamento .....	229
<b>7</b>	<b>Interfaces .....</b>	<b>233</b>
7.1	Interfaces como contratos .....	233
7.2	Tipos interface .....	237
7.3	Como satisfazer uma interface .....	238
7.4	Fazendo parse de flags com flag.Value .....	243
7.5	Valores interface .....	246
7.5.1	Ressalva: uma interface contendo um ponteiro nil não é nil.....	250
7.6	Ordenação com sort.Interface.....	251
7.7	A interface http.Handler .....	258
7.8	Interface error.....	264
7.9	Exemplo: avaliador de expressões .....	266
7.10	Asserções de tipo .....	275
7.11	Diferenciando erros com asserções de tipo.....	277
7.12	Consultando comportamentos com asserções de tipo interface.....	279
7.13	Switches de tipo .....	282
7.14	Exemplo: decodificação de XML baseada em token.....	285
7.15	Alguns conselhos.....	289
<b>8</b>	<b>Gorrotinas e canais.....</b>	<b>291</b>
8.1	Gorrotinas .....	292
8.2	Exemplo: Servidor de relógio concorrente .....	293
8.3	Exemplo: Servidor de eco concorrente.....	298
8.4	Canais.....	301
8.4.1	Canais sem buffer .....	302
8.4.2	Pipelines .....	304
8.4.3	Canais unidirecionais .....	307
8.4.4	Canais com buffer.....	309
8.5	Looping em paralelo.....	312
8.6	Exemplo: Web crawler concorrente .....	319

8.7	Multiplexando com select .....	324
8.8	Exemplo: Travessia concorrente de diretório .....	328
8.9	Cancelamento .....	333
8.10	Exemplo: Servidor de bate-papo .....	337
<b>9</b>	<b>Concorrência com variáveis compartilhadas .....</b>	<b>341</b>
9.1	Condições de concorrência .....	341
9.2	Exclusão mútua: sync.Mutex .....	348
9.3	Mutexes de leitura/escrita: sync.RWMutex .....	353
9.4	Sincronização de memória .....	354
9.5	Inicialização lazy: sync.Once .....	356
9.6	O detector de concorrência .....	359
9.7	Exemplo: Cache concorrente não bloqueante .....	360
9.8	Gorrotinas e threads .....	370
9.8.1	Pilhas que podem aumentar .....	370
9.8.2	Escalonamento de gorrotinas .....	371
9.8.3	GOMAXPROCS .....	372
9.8.4	Gorrotinas não têm identidade .....	373
<b>10</b>	<b>Pacotes e a ferramenta go .....</b>	<b>374</b>
10.1	Introdução .....	374
10.2	Caminhos de importação .....	375
10.3	A declaração do pacote .....	376
10.4	Declarações de importação .....	377
10.5	Importações vazias .....	379
10.6	Pacotes e nomenclatura .....	381
10.7	Ferramenta go .....	383
10.7.1	Organização do workspace .....	384
10.7.2	Fazendo download de pacotes .....	386
10.7.3	Build de pacotes .....	388
10.7.4	Documentando pacotes .....	391
10.7.5	Pacotes internos .....	393
10.7.6	Consulta de pacotes .....	394
<b>11</b>	<b>Testes .....</b>	<b>397</b>
11.1	A ferramenta go test .....	398
11.2	Funções Test .....	399
11.2.1	Testes aleatórios .....	404
11.2.2	Testando um comando .....	406
11.2.3	Testes caixa-branca .....	409
11.2.4	Pacotes de testes externos .....	413

11.2.5	Escrevendo testes eficazes.....	415
11.2.6	Evitando testes frágeis.....	417
11.3	Cobertura .....	418
11.4	Funções Benchmark.....	421
11.5	Profiling.....	425
11.6	Funções Example.....	428
<b>12</b>	<b>Reflexão .....</b>	<b>431</b>
12.1	Por que usar reflexão? .....	431
12.2	reflect.Type e reflect.Value .....	433
12.3	Display: um printer recursivo de valores.....	436
12.4	Exemplo: Codificando expressões-S .....	442
12.5	Atualizando variáveis com reflect.Value.....	446
12.6	Exemplo: Decodificando expressões-S.....	449
12.7	Acessando tags de campos de estrutura.....	454
12.8	Exibindo os métodos de um tipo .....	458
12.9	Uma advertência .....	459
<b>13</b>	<b>Programação de baixo nível.....</b>	<b>461</b>
13.1	unsafe.Sizeof, Alignof e Offsetof.....	462
13.2	unsafe.Pointer .....	465
13.3	Exemplo: Equivalência profunda .....	468
13.4	Chamando código C com cgo .....	471
13.5	Outra advertência .....	478