

Estruturas de dados e algoritmos em JavaScript

Escreva um código JavaScript complexo e eficaz
usando a mais recente ECMAScript

2ª Edição

Loiane Groner

Packt

Novatec

Copyright © Packt Publishing 2018. First published in the English language under the title '*Learning JavaScript Data Structures and Algorithms - Third Edition* – (9781788623872)'

Copyright © Packt Publishing 2018. Publicação original em inglês intitulada '*Learning JavaScript Data Structures and Algorithms - Third Edition* – (9781788623872)'

Esta tradução é publicada e vendida com a permissão da Packt Publishing.

© Novatec Editora Ltda. [2018].

Todos os direitos reservados e protegidos pela Lei 9.610 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

Tradução: Lúcia A. Kinoshita

Revisão gramatical: Tássia Carvalho

Editoração eletrônica: Carolina Kuwabata

ISBN: 978-85-7522-693-3

Histórico de impressões:

Fevereiro/2019 Segunda edição

Abril/2018 Primeira reimpressão

Março/2017 Primeira edição (ISBN: 978-85-7522-553-0)

Novatec Editora Ltda.

Rua Luís Antônio dos Santos 110

02460-000 – São Paulo, SP – Brasil

Tel.: +55 11 2959-6529

E-mail: novatec@novatec.com.br

Site: www.novatec.com.br

Twitter: twitter.com/novateceditora

Facebook: facebook.com/novatec

LinkedIn: linkedin.com/in/novatec

PY20190221

Sumário

Colaboradores.....	17
Prefácio	19
Capítulo 1 ■ JavaScript – uma visão geral rápida	24
Estrutura de dados e algoritmos em JavaScript.....	25
Configurando o ambiente.....	25
Configuração mínima para trabalhar com JavaScript	26
Usando servidores web.....	27
http-server do Node.js.....	29
Básico sobre o JavaScript.....	30
Variáveis	31
Escopo das variáveis.....	33
Operadores.....	34
Verdadeiro e falso.....	38
Funções dos operadores de igualdade (== e ===).....	39
Estruturas de controle	41
Instruções condicionais.....	41
Laços.....	43
Funções	44
Programação orientada a objetos em Javascript	45
Depuração e ferramentas.....	47
Depuração com o VSCode.....	48
Resumo	49
Capítulo 2 ■ Visão geral sobre ECMAScript e TypeScript.....	50
ECMAScript ou JavaScript?.....	50
ES6, ES2015, ES7, ES2016, ES8, ES2017 e ES.Next.....	51
Tabela de compatibilidade.....	51
Usando o Babel.js.....	53
Funcionalidades das versões ECMAScript 2015+	53

let e const no lugar de var	54
Escopo de variáveis com let e const.....	55
Templates literais	57
Funções de seta.....	58
Valores default para parâmetros de funções.....	59
Declarando os operadores de espalhamento e rest	60
Propriedades melhoradas de objetos	61
Programação orientada a objetos com classes.....	62
Herança.....	63
Trabalhando com getters e setters.....	64
Operador de exponencial	65
Módulos.....	66
Executando módulos ES2015 no navegador e com o Node.js.....	68
Usando importações nativas da ES2015 no Node.js.....	69
Executando módulos ES2015 no navegador	70
Compatibilidade de versões anteriores a ES2015+	72
Introdução ao TypeScript.....	72
Inferência de tipo	74
Interfaces.....	74
Genéricos	76
Outras funcionalidades do TypeScript.....	77
Verificações do TypeScript em tempo de compilação em arquivos JavaScript.....	77
Resumo	78

Capítulo 3 ■ Arrays..... 79

Por que devemos usar arrays?	79
Criando e inicializando arrays.....	80
Acessando elementos e fazendo uma iteração em um array.....	81
Acrescentando elementos	82
Inserindo um elemento no final do array.....	82
Usando o método push	82
Inserindo um elemento na primeira posição	83
Usando o método unshift.....	83
Removendo elementos.....	84
Removendo um elemento do final do array	84
Removendo um elemento da primeira posição.....	84
Usando o método shift	86
Adicionando e removendo elementos de uma posição específica	86
Arrays bidimensionais e multidimensionais.....	87
Iterando pelos elementos de arrays bidimensionais.....	88
Arrays multidimensionais.....	89

Referências para métodos de array em JavaScript	90
Juntando vários arrays	91
Funções de iteração	92
Iterando com o método every	92
Iterando com o método some	93
Iterando com forEach	93
Usando map e filter	93
Usando o método reduce	94
ECMAScript 6 e as novas funcionalidades de array	94
Iterando com o laço for..of	95
Usando o objeto @@iterator	95
Métodos entries, keys e values de array	96
Usando o método from	97
Usando o método Array.of	97
Usando o método fill	98
Usando o método copyWithin	99
Ordenando elementos	99
Ordenação personalizada	100
Ordenando strings	101
Pesquisa	102
ECMAScript 2015 – os métodos find e findIndex	103
ECMAScript 2016 – usando o método includes	103
Convertendo um array em uma string	104
Classe TypedArray	104
Arrays em TypeScript	105
Resumo	107

Capítulo 4 ■ Pilhas 108

Criação de uma biblioteca de estruturas de dados e algoritmos JavaScript	108
Estrutura de dados de pilha	110
Criando uma classe Stack baseada em array	111
Push de elementos na pilha	112
Pop de elementos da pilha	112
Dando uma espiada no elemento que está no topo da pilha	112
Verificando se a pilha está vazia	113
Limpando os elementos da pilha	114
Usando a classe Stack	114
Criando uma classe JavaScript Stack baseada em objeto	116
Push de elementos na pilha	117
Verificando se a pilha está vazia e o seu tamanho	117
Pop de elementos da pilha	118

Dando uma espiada no topo e limpando a pilha	119
Criando o método toString.....	119
Protegendo os elementos internos da estrutura de dados	120
Convenção de nomenclatura com underscore.....	121
Classes ES2015 com símbolos no escopo	121
Classes ES2015 com WeakMap	122
Proposta para campos de classe na ECMAScript	123
Resolvendo problemas usando pilhas.....	124
Convertendo números decimais para binários.....	125
Algoritmo conversor de base.....	126
Resumo	127

Capítulo 5 ■ Filas e deque 128

Estrutura de dados de fila.....	128
Criando a classe Queue.....	129
Inserção de elementos na fila	130
Remoção de elementos da fila	131
Dando uma espiada no elemento que está na frente da fila.....	132
Verificando se a pilha está vazia e o seu tamanho	132
Limpando a fila.....	133
Criando o método toString.....	133
Usando a classe Queue.....	134
Estrutura de dados de deque	135
Criando a classe Deque	136
Adicionando elementos na frente do deque	136
Usando a classe Deque	138
Resolvendo problemas usando filas e deque.....	139
Fila circular – Batata Quente	139
Verificador de palíndromo.....	141
Filas de tarefas em JavaScript.....	142
Resumo	143

Capítulo 6 ■ Listas ligadas 144

Estrutura de dados da lista ligada	144
Criando a classe LinkedList	146
Inserindo elementos no final da lista ligada	148
Removendo elementos de uma posição específica da lista ligada	150
Percorrendo a lista com um laço até alcançar a posição desejada.....	152
Refatorando o método remove.....	153

Inserindo um elemento em qualquer posição.....	154
Método indexOf: devolvendo a posição de um elemento.....	156
Removendo um elemento da lista ligada.....	157
Métodos isEmpty, size e getHead	157
Método toString.....	158
Listas duplamente ligadas.....	159
Inserindo um novo elemento em qualquer posição	160
Removendo elementos de qualquer posição.....	163
Listas ligadas circulares.....	166
Inserindo um novo elemento em qualquer posição	167
Removendo elementos de qualquer posição.....	169
Listas ligadas ordenadas.....	170
Inserindo elementos na ordem	171
Criando a classe StackLinkedList	172
Resumo	174

Capítulo 7 ■ Conjuntos 175

Estruturando um conjunto de dados.....	175
Criando uma classe Set	176
Método has(element)	177
Método add	178
Métodos delete e clear	178
Método size	179
Método values	180
Usando a classe Set	181
Operações em conjuntos	182
União de conjuntos	182
Intersecção de conjuntos	185
Aperfeiçoando o método intersection	186
Diferença entre conjuntos.....	187
Subconjunto.....	189
ECMAScript 2015 – a classe Set	191
Operações com a classe Set da ES2015.....	192
Simulando a operação de união	192
Simulando a operação de intersecção	192
Simulando a operação de diferença	193
Usando o operador de espalhamento	194
Multiconjuntos ou bags	195
Resumo	195

Capítulo 8 ■ Dicionários e hashes..... 196

Estrutura de dados de dicionário	196
Criando a classe Dictionary.....	197
Verificando se uma chave está presente no dicionário	199
Definindo uma chave e um valor no dicionário, e a classe ValuePair.....	200
Removendo um valor do dicionário	201
Obtendo um valor do dicionário.....	201
Métodos keys, values e valuePairs	202
Iterando pelos ValuePairs do dicionário com forEach.....	204
Métodos clear, size, isEmpty e toString	205
Usando a classe Dictionary.....	206
Tabela hash.....	207
Criando uma classe HashTable	208
Criando uma função de hash.....	209
Inserindo uma chave e um valor na tabela hash.....	210
Obtendo um valor da tabela hash	210
Removendo um valor da tabela hash.....	211
Usando a classe HashTable	212
Tabela hash versus conjunto hash	213
Tratando colisões nas tabelas hash	213
Encadeamento separado	215
Método put.....	216
Método get	217
Método remove	218
Sondagem linear	219
Método put.....	222
Método get	224
Método remove.....	225
Criando funções melhores de hash	227
Classe Map da ES2015	228
Classes WeakMap e WeakSet da ES2015	229
Resumo	230

Capítulo 9 ■ Recursão.....231

Entendendo a recursão	231
Calculando o fatorial de um número.....	232
Fatorial iterativo.....	233
Fatorial recursivo	233
Pilha de chamadas	234
Limitação do tamanho da pilha de chamadas em JavaScript	236

Sequência de Fibonacci	237
Fibonacci iterativo	237
Fibonacci recursivo	237
Fibonacci com memoização	238
Por que usar recursão? É mais rápido?	239
Resumo	239

Capítulo 10 ■ Árvores 240

Estrutura de dados de árvore	240
Terminologia de árvores	241
Árvore binária e árvore binária de busca	242
Criando as classes Node e BinarySearchTree	243
Inserindo uma chave na BST	244
Percorrendo uma árvore	248
Percurso em-ordem	248
Percurso pré-ordem	250
Percurso pós-ordem	251
Pesquisando valores em uma árvore	252
Pesquisando valores mínimos e máximos	252
Pesquisando um valor específico	254
Removendo um nó	257
Removendo uma folha	258
Removendo um nó com um filho à esquerda ou à direita	259
Removendo um nó com dois filhos	260
Árvores autobalanceadas	261
Árvore de Adelson-Velskii e Landi (árvore AVL)	262
Altura de um nó e o fator de balanceamento	263
Operações de balanceamento – rotações na árvore AVL	265
Rotação Esquerda-Esquerda: rotação simples à direita	265
Rotação Direita-Direita: rotação simples à esquerda	266
Esquerda-Direita: rotação dupla à direita	267
Direita-Esquerda: rotação dupla à esquerda	269
Inserindo um nó na árvore AVL	270
Removendo um nó da árvore AVL	272
Árvore rubro-negra	273
Inserindo um nó na árvore rubro-negra	274
Verificando as propriedades da árvore rubro-negra após a inserção	276
Rotações na árvore rubro-negra	280
Resumo	282

Capítulo 11 ■ Heap binário e heap sort..... 283

Estrutura de dados do heap binário	283
Criando a classe MinHeap	284
Representação da árvore binária com um array	285
Inserindo um valor no heap	286
Operação de sift up	286
Encontrando os valores mínimo e máximo no heap.....	288
Extraindo os valores mínimo e máximo do heap	289
Operação de sift down	290
Criando a classe MaxHeap.....	292
Algoritmo de heap sort.....	292
Resumo	295

Capítulo 12 ■ Grafos..... 296

Terminologia dos grafos	296
Grafos direcionados e não direcionados	298
Representando um grafo	299
A matriz de adjacências	299
Lista de adjacências.....	300
Matriz de incidências	301
Criando a classe Graph	301
Percorrendo grafos	305
Busca em largura (BFS)	307
Encontrando os caminhos mais curtos usando BFS.....	309
Estudos adicionais sobre algoritmos de caminhos mais curtos	312
Busca em profundidade (DFS)	313
Explorando o algoritmo DFS	316
Ordenação topológica usando DFS	319
Algoritmos de caminho mais curto	321
Algoritmo de Dijkstra.....	322
Algoritmo de Floyd-Warshall.....	325
Árvore de extensão mínima (MST)	327
Algoritmo de Prim	327
Algoritmo de Kruskal.....	329
Resumo	331

Capítulo 13 ■ Algoritmos de ordenação e de busca332

Algoritmos de ordenação.....	332
Bubble sort	333
Bubble sort melhorado	335

Selection sort	336
Insertion sort	338
Merge sort	340
Quick sort	343
Processo de partição.....	344
Quick sort em ação	345
Counting sort	348
Bucket sort.....	351
Radix sort.....	353
Algoritmos de busca.....	356
Busca sequencial	356
Busca binária	357
Busca por interpolação	359
Algoritmos de embaralhamento	361
Algoritmo de embaralhamento de Fisher-Yates	362
Resumo	363

Capítulo 14 ■ Designs de algoritmos e técnicas..... 364

Dividir e conquistar	364
Busca binária	365
Programação dinâmica.....	367
Problema do número mínimo de moedas para troco.....	368
Problema da mochila.....	371
Maior subsequência comum	374
Multiplificação de cadeia de matrizes.....	376
Algoritmos gulosos	379
Problema do número mínimo de moedas para troco.....	379
Problema fracionário da mochila	381
Algoritmos de backtracking.....	382
Rato em um labirinto	383
Solucionador de sudoku	385
Introdução à programação funcional	389
Programação funcional versus programação imperativa.....	389
ES2015+ e a programação funcional.....	391
Caixa de ferramentas funcional de JavaScript – map, filter e reduce	392
Bibliotecas e estruturas de dados funcionais de JavaScript	395
Resumo	395

Capítulo 15 ■ Complexidade de algoritmos	396
Notação big-O	396
Compreendendo a notação big-O	397
O(1).....	397
O(n)	397
O(n ²)	398
Comparando as complexidades	400
Estruturas de dados	401
Grafos	401
Algoritmos de ordenação.....	401
Algoritmos de busca.....	402
Introdução à teoria de NP-completo	402
Problemas impossíveis e algoritmos heurísticos	404
Divertindo-se com algoritmos.....	404
Resumo	406