

Auf einen Blick

| | |
|--|------------|
| Über die Autoren | 11 |
| Einführung | 25 |
| Teil I: Erste Schritte | 31 |
| Kapitel 1: Grundlegendes über Algorithmen | 31 |
| Kapitel 2: Algorithmen-Design | 45 |
| Kapitel 3: Mit Python Algorithmen verwenden | 63 |
| Kapitel 4: Algorithmen mit Python programmieren: Grundlagen | 85 |
| Kapitel 5: Grundlagen der Datenbearbeitung mit Python | 107 |
| Teil II: Die Notwendigkeit des Suchens und Sortierens | 127 |
| Kapitel 6: Daten strukturieren | 129 |
| Kapitel 7: Daten ordnen und durchsuchen | 145 |
| Teil III: Die Welt der Graphen | 165 |
| Kapitel 8: Die Grundlagen von Graphen | 167 |
| Kapitel 9: Punkte verbinden | 183 |
| Kapitel 10: Die Geheimnisse der Graphen | 207 |
| Kapitel 11: Die richtige Webseite finden | 217 |
| Teil IV: Der Kampf mit Big Data | 233 |
| Kapitel 12: Big Data verwalten | 235 |
| Kapitel 13: Abläufe parallelisieren | 257 |
| Kapitel 14: Daten komprimieren | 271 |
| Teil V: Komplexe Aufgaben angehen | 285 |
| Kapitel 15: Mit gierigen Algorithmen arbeiten | 287 |
| Kapitel 16: Dynamische Programmierung | 301 |
| Kapitel 17: Randomisierte Algorithmen | 323 |
| Kapitel 18: Lokale Suchen durchführen | 339 |
| Kapitel 19: Lineare Optimierung | 355 |
| Kapitel 20: Heuristiken untersuchen | 367 |
| Teil VI: Zehn große Algorithmen | 383 |
| Kapitel 21: Zehn algorithmische Verfahren, die die Welt verändern | 385 |
| Kapitel 22: Zehn bislang ungelöste Probleme | 391 |
| Stichwortverzeichnis | 397 |

Inhaltsverzeichnis

| | |
|--|-----------|
| Über die Autoren | 11 |
| John Muellers Widmung | 11 |
| Luca Massarons Widmung | 11 |
| John Muellers Danksagung | 12 |
| Luca Massarons Danksagung | 12 |
| Einführung | 25 |
| Über dieses Buch | 25 |
| Konventionen in diesem Buch | 26 |
| Törichte Annahmen über den Leser | 26 |
| Symbole, die in diesem Buch verwendet werden | 27 |
| Wie es weitergeht | 28 |
| TEIL I | |
| ERSTE SCHRITTE | 31 |
| Kapitel 1 | |
| Grundlegendes über Algorithmen | 31 |
| Algorithmen beschreiben | 32 |
| Definitionen zur Anwendung von Algorithmen | 33 |
| Algorithmen sind überall | 36 |
| Mit Computern Aufgaben lösen | 37 |
| Moderne CPUs und GPUs wirksam einsetzen | 37 |
| Arbeiten mit Spezialchips | 38 |
| Netzwerke wirksam einsetzen | 39 |
| Daten effektiv nutzen | 39 |
| Zwischen Aufgaben und Lösungen unterscheiden | 40 |
| Richtigkeit und Effizienz | 40 |
| Die Erkenntnis, dass nichts umsonst ist | 41 |
| Die Strategie an die Aufgabe anpassen | 41 |
| Algorithmen in einer Lingua franca beschreiben | 41 |
| Schwierige Aufgaben angehen | 42 |
| Daten für Lösungen strukturieren | 42 |
| Die Sichtweise eines Computers | 42 |
| Datenordnung muss sein | 43 |
| Kapitel 2 | |
| Algorithmendesign | 45 |
| Der Anfang der Problemlösung | 46 |
| Praxisaufgaben modellieren | 46 |

16 Inhaltsverzeichnis

| | |
|---|-----------|
| Lösungen und Gegenbeispiele finden | 48 |
| Auf den Schultern von Riesen stehen | 49 |
| Teile und herrsche | 49 |
| Brute-Force-Lösungen vermeiden | 50 |
| Erster Schritt: Vereinfachen | 51 |
| Reduktion ist meist vorteilhaft | 51 |
| Erkennen, dass Gier gut sein kann | 52 |
| Gierig denken | 52 |
| Eine gute Lösung | 53 |
| Kostenberechnungen und Heuristiken | 54 |
| Das Problem in einem Raum darstellen | 54 |
| Zufällig und von Glück gesegnet | 55 |
| Heuristiken und Kostenfunktionen verwenden | 55 |
| Algorithmen auswerten | 56 |
| Simulationen mittels abstrakter Maschinen | 57 |
| Weitere Abstrahierungen | 58 |
| Mit Funktionen arbeiten | 59 |
| Kapitel 3 | |
| Mit Python Algorithmen verwenden | 63 |
| Die Vorteile von Python | 64 |
| Wie Python in diesem Buch verwendet wird | 65 |
| Mit MATLAB arbeiten | 66 |
| Andere Testumgebungen für Algorithmen | 67 |
| Python-Distributionen | 67 |
| Analytics Anaconda installieren | 68 |
| Enthought Canopy Express | 69 |
| Python(x,y) | 69 |
| WinPython | 69 |
| Python auf Linux installieren | 70 |
| Python auf MacOS installieren | 71 |
| Python auf Windows installieren | 72 |
| Datensätze und Beispielcode herunterladen | 76 |
| Jupyter Notebook benutzen | 76 |
| Das Coderepository definieren | 77 |
| Die Datensätze in diesem Buch verstehen | 83 |
| Kapitel 4 | |
| Algorithmen mit Python programmieren: Grundlagen | 85 |
| Mit Zahlen und Logik arbeiten | 86 |
| Variablenzuordnungen vornehmen | 87 |
| Berechnungen durchführen | 88 |
| Datenvergleich durch boolesche Ausdrücke | 89 |
| Strings erstellen und verwenden | 91 |
| Mit Datumsangaben arbeiten | 92 |
| Funktionen erstellen und verwenden | 93 |

Inhaltsverzeichnis 17

| | |
|---|-----|
| Mehrfach aufrufbare Funktionen erstellen | 93 |
| Funktionen aufrufen | 94 |
| Bedingte Anweisungen und Schleifen verwenden | 97 |
| Entscheidungen mittels if-Befehl treffen | 97 |
| Mittels verschachtelter Entscheidungen zwischen mehreren Möglichkeiten auswählen | 98 |
| Iterative Aufgaben mittels for-Schleife ausführen | 99 |
| Der while-Befehl | 100 |
| Daten in Mengen, Katalogen und Tupeln speichern | 101 |
| Datasets erstellen | 101 |
| Listen erstellen | 102 |
| Tupel erstellen und verwenden | 103 |
| Nützliche Iteratoren definieren | 104 |
| Daten mittels Dictionaries indizieren | 105 |

**Kapitel 5
Grundlagen der Datenbearbeitung mit Python 107**

| | |
|--|-----|
| Berechnungen mit Vektoren und Matrizen | 108 |
| Variablen- und Vektoroperationen verstehen | 108 |
| Vektoren multiplizieren | 110 |
| Der erste Schritt: Matrizen erstellen | 111 |
| Matrizen multiplizieren | 112 |
| Erweiterte Matrizenoperationen definieren | 114 |
| Der richtige Weg: Kombinationen erzeugen | 115 |
| Permutationen unterscheiden | 116 |
| Kombinationen erzeugen | 117 |
| Wiederholungen behandeln | 118 |
| Ergebnisse mit Rekursion erzielen | 118 |
| Die Funktionsweise der Rekursion | 118 |
| Endrekursionen entfernen | 121 |
| Prozesse schneller ausführen | 122 |
| Der »teile und herrsche«-Ansatz | 122 |
| Zwischen möglichen Lösungen unterscheiden | 125 |

**TEIL II
DIE NOTWENDIGKEIT DES SUCHENS UND SORTIERENS 127**

**Kapitel 6
Daten strukturieren 129**

| | |
|--|-----|
| Die Notwendigkeit einer Struktur | 130 |
| Inhalt vereinfacht darstellen | 130 |
| Daten aus verschiedenen Quellen anpassen | 131 |
| Die Wichtigkeit der Datenbereinigung | 132 |
| Daten gestapelt und kumuliert anordnen | 134 |
| Stapelweise Anordnungen | 135 |

18 Inhaltsverzeichnis

| | |
|--|-----|
| Warteschlangen verwenden | 136 |
| Daten mittels assoziativen Datenfeldern finden | 137 |
| Mit Bäumen arbeiten | 138 |
| Grundlegendes über Bäume | 139 |
| Bäume erstellen | 139 |
| Relationen in Graphen darstellen | 141 |
| Über Bäume hinauswachsen | 142 |
| Graphen erstellen | 143 |

Kapitel 7

Daten ordnen und durchsuchen 145

| | |
|--|-----|
| Daten mit Mergesort und Quicksort sortieren | 146 |
| Die Notwendigkeit des Sortierens | 146 |
| Daten naiv anordnen | 147 |
| Bessere Sortiertechniken einsetzen | 149 |
| Suchbäume und Heaps verwenden | 153 |
| Die Notwendigkeit einer effizienten Suche | 154 |
| Binäre Suchbäume erstellen | 156 |
| Spezialisierte Suchen mit einem binären Heap | 157 |
| Hashing verwenden | 158 |
| Alles in Buckets füllen | 158 |
| Kollisionen vermeiden | 160 |
| Hashfunktionen selbst erstellen | 161 |

TEIL III

DIE WELT DER GRAPHEN 165

Kapitel 8

Die Grundlagen von Graphen 167

| | |
|---|-----|
| Die Wichtigkeit von Netzwerken | 168 |
| Grundlegendes über Graphen | 168 |
| Graphen sind überall | 170 |
| Die soziale Seite eines Graphen | 171 |
| Teilgraphen verstehen | 172 |
| Definitionen für das Zeichnen von Graphen | 172 |
| Die wichtigsten Eigenschaften von Graphen | 173 |
| Graphen zeichnen | 174 |
| Die Funktionalität eines Graphen | 175 |
| Kanten und Knoten zählen | 176 |
| Zentralität berechnen | 178 |
| Graphen in ein numerisches Format bringen | 180 |
| Graphen zu einer Matrix hinzufügen | 181 |
| Dünn besetzte Matrizen verwenden | 182 |
| Graphen durch Listen ausdrücken | 182 |

| | |
|---|------------|
| Kapitel 9 | |
| Punkte verbinden | 183 |
| Graphen effizient durchsuchen | 184 |
| Einen Graphen erstellen | 184 |
| Breitensuche anwenden | 186 |
| Tiefensuche anwenden | 187 |
| Die Entscheidung für eine Methode | 189 |
| Die Elemente eines Graphen sortieren | 189 |
| Mit gerichteten azyklischen Graphen arbeiten | 190 |
| Topologische Sortierungen verwenden | 191 |
| Die Reduktion auf einen minimalen Spannbaum | 192 |
| Geeignete Algorithmen verwenden | 195 |
| Vorrangwarteschlangen kennenlernen | 195 |
| Den Algorithmus von Prim einsetzen | 196 |
| Den Algorithmus von Kruskal testen | 198 |
| Entscheiden, welcher Algorithmus am besten passt | 200 |
| Den kürzesten Weg finden | 201 |
| Definieren, was der kürzeste Weg ist | 201 |
| Der Dijkstra-Algorithmus: eine Erklärung | 203 |
| Kapitel 10 | |
| Die Geheimnisse der Graphen | 207 |
| Soziale Netzwerke als Graphen betrachten | 207 |
| Netzwerke in Gruppen clustern | 208 |
| Communitys entdecken | 210 |
| Einen Graphen durchlaufen | 213 |
| Grade der Trennung abzählen | 213 |
| Graphen zufällig durchlaufen | 215 |
| Kapitel 11 | |
| Die richtige Webseite finden | 217 |
| Die Welt in einer Suchmaschine | 218 |
| Datensuche im Internet | 218 |
| Die richtigen Daten finden | 218 |
| Die Funktionsweise des PageRank-Algorithmus | 219 |
| Die Logik hinter dem PageRank-Algorithmus | 220 |
| Das A und O von PageRank | 221 |
| PageRank implementieren | 222 |
| Pythonskripte implementieren | 222 |
| Der Kampf mit naiven Implementierungen | 225 |
| Langeweile und Teleportation | 228 |
| Das Leben einer Suchmaschine | 229 |
| Andere Verwendungsmöglichkeiten von PageRank | 230 |
| Über das PageRank-Paradigma hinaus | 230 |
| Semantisches Suchen | 231 |
| Ranking von Suchergebnissen mit künstlicher Intelligenz | 231 |

20 Inhaltsverzeichnis

| | |
|---|------------|
| TEIL IV | |
| DER KAMPF MIT BIG DATA | 233 |
| Kapitel 12 | |
| Big Data verwalten | 235 |
| Die Umwandlung von Strom in Daten | 236 |
| Die Auswirkungen von Moore | 236 |
| Daten sind überall | 238 |
| Algorithmen ins Geschäft bringen | 240 |
| Datenflüsse streamen | 242 |
| Streams korrekt analysieren | 243 |
| Die richtigen Daten auswählen | 244 |
| Lösungen aus Stream-Daten skizzieren | 248 |
| Datenstromelemente filtern | 248 |
| Bloomfilter darstellen | 251 |
| Die Anzahl der Elemente bestimmen | 254 |
| Elemente eines Streams abzählen | 256 |
| Kapitel 13 | |
| Abläufe parallelisieren | 257 |
| Die Handhabung großer Datenmengen | 258 |
| Die parallele Methode | 258 |
| Dateien und Vorgänge verteilen | 260 |
| Lösungen mit MapReduce | 262 |
| Operationen verteilen | 264 |
| Algorithmen für MapReduce erstellen | 265 |
| MapReduce-Simulationen erstellen | 266 |
| Anfragen durch Mapping | 268 |
| Kapitel 14 | |
| Daten komprimieren | 271 |
| Daten reduzieren | 272 |
| Codierungen verstehen | 272 |
| Die Folgen der Kompression | 273 |
| Die Entscheidung für eine Kompressionsart | 275 |
| Die Wahl einer vernünftigen Codierung | 276 |
| Codieren mit der Huffman-Kompression | 279 |
| Folgen merken mit dem LZW-Algorithmus | 280 |

| | |
|--|------------|
| TEIL V | |
| KOMPLEXE AUFGABEN ANGEHEN | 285 |
| Kapitel 15 | |
| Mit gierigen Algorithmen arbeiten | 287 |
| Die Entscheidung, wann Gier Sinn macht. | 288 |
| Die Vorteile der Gier | 289 |
| Gierige Algorithmen im Zaum halten | 290 |
| NP-vollständige Probleme | 292 |
| Herausfinden, wieso Gier nützlich ist | 293 |
| Daten im Cache anordnen | 294 |
| Der Kampf um Ressourcen | 295 |
| Noch mal zu Huffman-Codierungen | 297 |
| Kapitel 16 | |
| Dynamische Programmierung | 301 |
| Dynamische Programmierung – was ist das? | 302 |
| Die historische Grundlage | 302 |
| Probleme dynamisieren | 303 |
| Rekursion dynamisch auslegen | 304 |
| Memoisation verwenden | 307 |
| Die besten dynamischen Techniken | 309 |
| Den Rucksack packen | 309 |
| Städte bereisen | 313 |
| Näherungsweise nach Strings suchen | 317 |
| Kapitel 17 | |
| Randomisierte Algorithmen | 323 |
| Die Funktionsweise der Randomisierung | 324 |
| Die Notwendigkeit der Randomisierung | 324 |
| Das Wahrscheinlichkeitsprinzip verstehen | 325 |
| Verteilungen verstehen | 326 |
| Die Monte-Carlo-Simulation | 329 |
| Den Zufall in die Logik einbauen | 331 |
| Den Median durch Quickselect bestimmen | 332 |
| Monte-Carlo-Simulationen ausführen | 334 |
| Schneller sortieren mit Quicksort | 337 |
| Kapitel 18 | |
| Lokale Suchen durchführen | 339 |
| Lokale Suchen verstehen | 340 |
| Die Nachbarschaft kennen | 340 |
| Tricks bei der lokalen Suche | 342 |
| Bergsteigen und das Damenproblem | 343 |

22 Inhaltsverzeichnis

| | |
|--|------------|
| Die Funktionsweise des Simulated Annealing | 346 |
| Wiederholungen durch die Tabu-Suche vermeiden | 347 |
| Die Erfüllbarkeit boolescher Schaltkreise | 348 |
| 2-SAT mittels Randomisierungen lösen | 349 |
| Die Code-Implementierung in Python | 350 |
| Die Wichtigkeit des Anfangspunkts | 353 |
| Kapitel 19 | |
| Lineare Optimierung | 355 |
| Lineare Funktionen – ein Werkzeug | 356 |
| Die mathematischen Grundlagen | 357 |
| Während des Planens vereinfachen | 358 |
| Das Simplex-Verfahren geometrisch bearbeiten | 359 |
| Die Grenzen der linearen Optimierung | 361 |
| Lineare Optimierung in der Praxis | 361 |
| PuLP auf dem eigenen Computer installieren | 362 |
| Produktionsvorgänge und Einnahmen optimieren | 362 |
| Kapitel 20 | |
| Heuristiken untersuchen | 367 |
| Unterschiedliche Heuristiken kennenlernen | 368 |
| Die Ziele heuristischer Verfahren | 368 |
| Von genetisch zu künstlich | 369 |
| Heuristische Routensuche bei Robotern | 370 |
| Unbekannte Gebiete erkunden | 370 |
| Entfernungsmessungen als Heuristiken verwenden | 372 |
| Pathfinding-Algorithmen verstehen | 373 |
| Einen Irrgarten erstellen | 373 |
| Die Suche nach dem besten Weg | 376 |
| Heuristische Bewegungen mit A* | 379 |
| TEIL VI | |
| ZEHN GROSSE ALGORITHMEN | 383 |
| Kapitel 21 | |
| Zehn algorithmische Verfahren, die die Welt verändern | 385 |
| Sortierverfahren einsetzen | 386 |
| Suchen durchführen | 386 |
| Mit Zufallszahlen eine neue Ordnung herstellen | 386 |
| Datenkompressionen vornehmen | 387 |
| Die Geheimhaltung von Daten | 387 |
| Datenbereiche ändern | 388 |
| Zusammenhänge erkennen | 388 |
| Muster in Daten erkennen | 389 |
| Mit Automatisierung und automatischen Antworten arbeiten | 390 |
| Eindeutige Identifizierungen erstellen | 390 |

| | |
|---|------------|
| Kapitel 22 | |
| Zehn bislang ungelöste Probleme | 391 |
| Textsuchen bearbeiten | 392 |
| Wörter unterscheiden | 392 |
| Herausfinden, wann eine Anwendung anhält | 392 |
| Einwegfunktionen erstellen und verwenden | 393 |
| Die Multiplikation großer Zahlen | 393 |
| Die Gleichverteilung von Ressourcen | 394 |
| Die Berechnungszeit der Editierdistanz reduzieren | 394 |
| Aufgaben schnell lösen | 395 |
| Das Paritätsspiel spielen | 395 |
| Räumliche Aufgabenstellungen verstehen | 395 |
| Stichwortverzeichnis | 397 |

