

ОГЛАВЛЕНИЕ

Кто мы.....	16
Наша философия.....	16
Наш опыт.....	17
Исходный код примеров.....	18
От издательства	19
Введение	20
О чем эта книга.....	20
Структура издания.....	21
Для кого эта книга.....	21
Ваше путешествие начинается.....	22

Часть I. Знакомство с Python и основные структуры данных

Глава 1. Введение в Python и алгоритмы.....	24
1.1. Назначение алгоритмов и структур данных.....	24
1.1.1. Значение эффективности.....	25
1.1.2. Организация данных	26
1.1.3. Гибкость и масштабируемость	26
1.1.4. Радость решения задач.....	27
1.1.5. Универсальность алгоритмов.....	27
1.1.6. Строительные блоки для расширенных концепций.....	28
1.1.7. Критическое мышление и навыки решения задач	28
1.1.8. Подготовка к техническим собеседованиям.....	29
1.1.9. Разные возможности.....	29

1.2. Эволюция программирования	30
1.2.1. Рассвет программирования: перфокарты и машинный код	30
1.2.2. Язык ассемблера и лестница абстракций	31
1.2.3. Языки высокого уровня: большой скачок	31
1.2.4. Структурная и объектно-ориентированная парадигмы.....	32
1.2.5. Современная эра: гибкость, открытый исходный код и Интернет.....	33
1.2.6. Будущее: квантовые вычисления, искусственный интеллект и не только	34
1.2.7. Интегрированные среды разработки и инструментарий.....	34
1.2.8. Движение за открытый исходный код.....	35
1.2.9. Мобильная революция и кросс-платформенная разработка	35
1.2.10. Облачные вычисления и бессерверные архитектуры.....	36
1.2.11. Контейнеры и микросервисы	37
1.2.12. Неавтоматизированные и автоматизированные платформы	37
1.3. Python и алгоритмы	38
1.3.1. Простой синтаксис Python: псевдокод оживает	39
1.3.2. Универсальность и библиотеки: сокровищница инструментов.....	40
1.3.3. Интерактивность с помощью Python: петля мгновенной обратной связи	41
1.3.4. Масштабируемость: от обучения к реальным решениям	42
1.3.5. Поддержка сообщества: пишем код вместе.....	42
1.3.6. Проблемы производительности и не только	42
1.3.7. Философское выравнивание: «Дзен Python» и алгоритмическое мышление	43
1.3.8. Адаптация: эволюция Python и современные алгоритмы	44
1.3.9. Красота разнообразия: многообразие парадигм и гибкость алгоритмов в Python	44
1.4. Роль Python в разработке алгоритмов	45
1.4.1. Доступная точка входа в Python: путь к алгоритмическому мышлению.....	46
1.4.2. Прототипирование: от идеи до воплощения	47
1.4.3. Визуализация и отладка: видеть — значит верить	47
1.4.4. Преодоление препятствия: перевод Python на другие языки	48
1.4.5. Создано для сотрудничества: совместное использование и развитие	49

1.4.6. Рост машинного обучения и искусственного интеллекта: Python на переднем крае	49
1.4.7. Интеграция с Си/C++: повышение производительности	50
1.4.8. Расширение Python с помощью алгоритмов: создание модулей и пакетов	51
1.4.9. Сообщество и открытый исходный код: стоя на плечах гигантов	51
Практические упражнения	52
Резюме	54
Глава 2. Погружение в Python	56
2.1. Основы синтаксиса Python	56
2.1.1. Отступы	56
2.1.2. Комментарии	57
2.1.3. Переменные	58
2.1.4. Операторы и выражения	58
2.1.5. Двоеточия	59
2.1.6. Функции	59
2.1.7. Списки и индексация	60
2.1.8. Обработка строк	61
2.1.9. Циклы	62
2.1.10. Словари	62
2.1.11. Обработка ошибок	63
2.2. Типы данных и операторы	64
2.2.1. Основные типы данных	64
2.2.2. Контейнеры	66
2.2.3. Операторы	68
2.3. Управляющие структуры и функции	74
2.3.1. Управляющие структуры	74
2.3.2. Вложенные управляющие структуры	77
2.3.3. Тернарный оператор	78
2.3.4. Лямбда-функции	78
2.3.5. Строки документации для функций	79
2.3.6. Рекурсия	80
2.3.7. Генераторы	80
Практические упражнения	81
Резюме	83

Глава 3. Простые контейнеры данных	85
3.1. Списки, кортежи, множества и словари	85
3.1.1. Списки	86
3.1.2. Кортежи	86
3.1.3. Множества	87
3.1.4. Словари	88
3.1.5. Генераторы списков	89
3.1.6. Распаковка кортежей	89
3.1.7. Операции с множествами	90
3.1.8. Словарные методы	91
3.2. Объектно-ориентированное программирование: классы, объекты и инкапсуляция	92
3.2.1. Классы и объекты	92
3.2.2. Инкапсуляция	93
3.2.3. Наследование	95
3.2.4. Полиморфизм	96
3.2.5. Композиция	97
3.2.6. Перегрузка методов	98
3.2.7. Цепочка методов	99
3.3. Стеки, очереди и их применение	100
3.3.1. Стеки	100
3.3.2. Очереди	102
3.3.3. Расширенное применение и вариации	104
3.4. Связанные списки: указатели, узлы и их применение	107
3.4.1. Что такое связанные списки	107
3.4.2. Основные компоненты	108
3.4.3. Типы связанных списков	109
3.4.4. Операции над связанными списками	110
3.4.5. Применение связанных списков	111
3.4.6. Преимущества связанных списков перед массивами	112
3.4.7. Недостатки связанных списков	113
3.4.8. Вариации на тему	114
3.4.9. Пример использования: управление памятью в операционных системах	114
3.4.10. Советы по работе со связанными списками	115
Практические упражнения	116
Резюме	118

Проект 1. Базовый калькулятор	120
Настройка основного фреймворка	120
Реализация арифметических функций	122
Интеграция арифметических функций с основным фреймворком	122
Улучшение пользовательского опыта.....	123
Добавление расширенных арифметических функций.....	124
Добавление расширенных функций.....	124
Функции памяти.....	125
Улучшение интерфейса пользователя и опыта взаимодействия с данным интерфейсом.....	126

Часть II. Сортировка, поиск и иерархические структуры

Глава 4. Сортировка	128
4.1. Основные алгоритмы сортировки	128
4.1.1. Пузырьковая сортировка	129
4.1.2. Сортировка выбором	130
4.1.3. Сортировка вставками	131
4.2. Расширенная сортировка.....	133
4.2.1. Быстрая сортировка: «разделяй и властвуй»	133
4.2.2. Сортировка слиянием: объединение упорядоченных списков.....	134
4.2.3. Пирамидальная сортировка: использование двоичной кучи	136
4.2.4. Сравнение расширенных алгоритмов сортировки.....	137
4.3. Анализ временной сложности и производительности	139
4.3.1. Концепция временной сложности	140
4.3.2. Нотация «О большое»	140
4.3.3. Не только временная сложность	141
4.3.4. Эмпирический анализ производительности	142
4.3.5. Практическое применение временной сложности	143
4.3.6. Инструменты визуализации	144
Практические упражнения	145
Резюме	147
Глава 5. Поисковые операции и эффективность	149
5.1. Линейный и бинарный поиск.....	149
5.1.1. Линейный поиск.....	150
5.1.2. Двоичный поиск.....	151

5.1.3. Сравнение алгоритмов	151
5.1.4. Анализ производительности	152
5.1.5. Применение в реальных сценариях	154
5.2. Хеширование	156
5.2.1. Что такое хеширование	156
5.2.2. Хеш-функция	157
5.2.3. Разработка хеш-функций	158
5.2.4. Эффективность хеширования	158
5.2.5. Применение хеширования	160
5.2.6. Криптографические хеш-функции	161
5.2.7. Встроенная в Python функция hash()	162
5.2.8. Потенциальные сложности	162
5.3. Хеш-таблицы: реализация и разрешение коллизий	163
5.3.1. Базовая реализация хеш-таблицы	164
5.3.2. Изменение размера хеш-таблицы	165
5.3.3. Методы разрешения коллизий	165
5.3.4. Обработка коллизий	165
5.3.5. Работа с удалениями	168
5.3.6. Применение и ограничения хеш-таблиц	169
5.3.7. Соображения безопасности	170
5.3.8. Сравнение хеш-таблицы, хеш-карты и словаря	170
5.4. Временная сложность и нотация «О большое»	171
5.4.1. Временная сложность	171
5.4.2. Нотация «О большое»	172
5.4.3. Важность анализа временной сложности	175
5.4.4. Визуализация нотаций «О большое»	176
5.4.5. Распространенные заблуждения и ошибки	177
5.4.6. Оценка поисковых алгоритмов с помощью нотации «О большое»	179
Практические упражнения	180
Резюме	182
Глава 6. Деревья и графы: иерархические структуры данных	185
6.1. Деревья: типы и методы обхода	185
6.1.1. Виды деревьев	186
6.1.2. Методы обхода деревьев	187

6.1.3. Расширенные концепции обхода.....	191
6.1.4. Практическое применение деревьев	191
6.2. Графы: представление и основные алгоритмы	193
6.2.1. Основные понятия.....	194
6.2.2. Представление графов.....	195
6.2.3. Основные алгоритмы построения графов.....	197
6.2.4. Практическое применение графов.....	199
6.2.5. Практические советы.....	200
Практические упражнения	200
Резюме.....	202
Тесты к части II.....	204
Проект 2. Приложение для списка контактов.....	206
Реализация базовой структуры	206
Определение контактного узла.....	206
Создание двоичного дерева поиска	207
Проверка базовой вставки	207
Добавление функции поиска.....	208
Добавление функции удаления.....	208
Перечисление всех контактов	209
Будущие улучшения.....	210

Часть III. Расширенные алгоритмические методы и сетевые структуры

Глава 7. Освоение алгоритмических методов	212
7.1. Стратегия «разделяй и властвуй»	212
7.1.1. Этапы.....	212
7.1.2. Преимущества.....	213
7.1.3. Применение.....	214
7.2. Динамическое программирование.....	216
7.2.1. Как работает метод	216
7.2.2. Подходы	217
7.2.3. Расширенные концепции	217
7.2.4. Применение.....	218

7.2.5. Сравнение динамического программирования и стратегии «разделяй и властвуй»	222
7.3. Жадный подход и поиск с возвратом.....	223
7.3.1. Жадный подход	223
7.3.2. Поиск с возвратом	227
7.3.3. Анализ сложности жадных алгоритмов и алгоритмов поиска с возвратом.....	229
Практические упражнения	230
Резюме.....	232
Глава 8. Сети и пути: расширенные графовые алгоритмы.....	234
8.1. Глубокое погружение в теорию графов.....	234
8.1.1. Темы повышенной сложности	235
8.1.2. Продвинутое алгоритмы.....	237
8.1.3. Применение в реальных задачах	243
8.2. Алгоритмы для поиска кратчайших путей, потоков и связности.....	245
8.2.1. Алгоритмы поиска кратчайшего пути	245
8.2.2. Алгоритмы сетевых потоков	248
8.2.3. Алгоритмы связности графов.....	251
8.3. Оптимизация сетей и передовые графовые методы.....	253
8.3.1. Оптимизация сетей	254
8.3.2. Продвинутое графовые методы	256
8.3.3. Кластеризация графов	257
8.3.4. Встраивание графов и анализ сетей	259
8.3.5. Анализ графов и большие данные.....	260
Практические упражнения	262
Резюме.....	264
Тесты к части III.....	266
Проект 3. Приложение для прокладки маршрутов на основе карты.....	268
Настройка графа для карты.....	268
Реализация алгоритма Дейкстры	269
Взаимодействие с пользователем и обработка ввода	270
Работа с картографическими данными реального мира.....	270
Графический интерфейс для визуализации (необязательно).....	271
Чего мы достигли	271

**Часть IV. Обработка строк, расширенные концепции
и практическое применение**

Глава 9. Расшифровка строк и образцов	274
9.1. Строковые алгоритмы.....	274
9.1.1. Области применения	275
9.1.2. Ключевые понятия	276
9.1.3. Дополнительные сведения.....	278
9.1.4. Расширенные методы обработки строк.....	281
9.2. Поиск по образцу, префиксные и суффиксные деревья	285
9.2.1. Поиск по образцу	285
9.2.2. Префиксные деревья	287
9.2.3. Суффиксные деревья.....	289
9.2.4. Дополнительные варианты использования.....	291
9.3. Передовые методы сопоставления с образцом и анализа текста	294
9.3.1. Расширенные методы работы с регулярными выражениями	295
9.3.2. Приближенное сопоставление строк (нечеткое сопоставление).....	297
9.3.3. Интеллектуальный анализ текста и аналитика.....	299
9.3.4. Обработка естественного языка и интеграция искусственного интеллекта	301
Практические упражнения	305
Резюме	307
Глава 10. Сложные вычислительные задачи	309
10.1. NP-трудные и NP-полные классы	309
10.1.1. NP-полнота	309
10.1.2. NP-трудность	311
10.1.3. Более широкие последствия для информатики.....	312
10.2. Аппроксимирующие и рандомизированные алгоритмы	315
10.2.1. Аппроксимирующие алгоритмы	315
10.2.2. Рандомизированные алгоритмы	318
10.2.3. Алгоритмы Монте-Карло и Лас-Вегас.....	320
10.2.4. Использование аппроксимирующих и рандомизированных алгоритмов	321
10.3. Сложные алгоритмы в теории графов и анализе сетей.....	322
10.3.1. Алгоритмы разбиения и кластеризации графов.....	323
10.3.2. Обработка графов.....	325

10.3.3. Расширенный сетевой поток и возможности подключения.....	326
10.3.4. Новые тенденции и современные приложения	328
Практические упражнения	332
Резюме	334
Глава 11. От теории к практике. Практические примеры и оптимизация	336
11.1. Практические примеры. Реальные алгоритмические решения	336
11.1.1. Оптимизация поисковой системы.....	336
11.1.2. Оптимизация цепочки поставок	338
11.1.3. Персонализированная медицина	340
11.1.4. Системы рекомендаций электронной коммерции	341
11.1.5. Оптимизация управления дорожным движением и маршрутизацией	342
11.1.6. Повышение безопасности сети с помощью систем обнаружения вторжений	344
11.1.7. Междоменные алгоритмические инновации	345
11.1.8. Важность алгоритмической этики и ответственного отношения к ИИ.....	345
11.2. Улучшение производительности Python.....	346
11.2.1. Факторы, влияющие на производительность Python.....	346
11.2.2. Методы повышения производительности Python.....	348
11.2.3. Передовые методы оптимизации и рекомендации	350
Практические упражнения	353
Резюме	356
Тесты к части IV.....	358
Проект 4. Система обнаружения плагиата	360
Предварительная обработка текста и измерение сходства.....	360
Работа с большими документами и анализ на уровне абзацев.....	362
Использование передовых методов анализа текста.....	363
Будущие улучшения.....	364
Заключение	365
Узнайте больше о нас.....	367