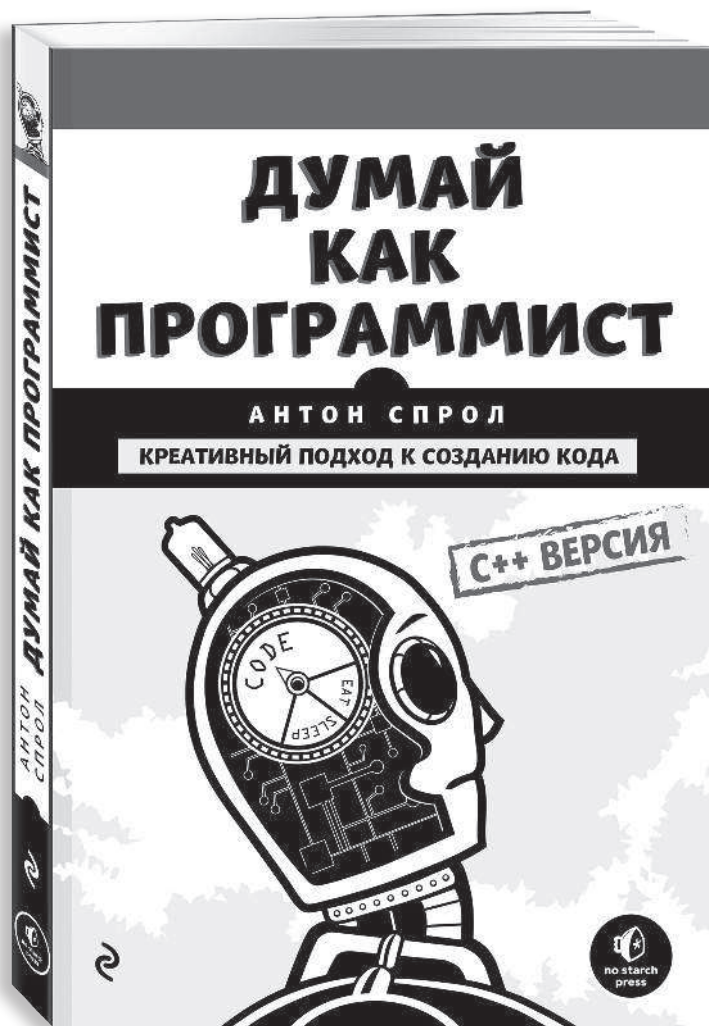


Научитесь решать задачи
программирования креативно!



- Живой язык
- Множество примеров на языке C++
- Уникальное авторское видение

ОГЛАВЛЕНИЕ

Об авторе	15
Введение	16
Для кого эта книга?	17
О книге	18
Как пользоваться этой книгой	19
Номера строк и отступов	19
Длинные строки кода	20
Загрузка и установка Python	21
Запуск интерактивной среды разработки	22
Поиск справочной документации	23
Глава 1. Интерактивная среда разработки	24
Немного арифметики	24
Целые числа и числа с плавающей запятой.....	25
Выражения	26
Вычисление выражений.....	26
Синтаксические ошибки.....	27
Сохранение значений в переменных	28
Заключение.....	33
Глава 2. Написание программ	34
Строковые значения	34
Конкатенация строк.....	35
Написание кода в среде разработки	36
Создание программы «Привет, мир!».....	37
Сохранение программы.....	38
Запуск программы.....	39
Как работает программа «Привет, мир!».....	40
Комментарии для программиста	40
Функции: Мини-программы внутри программ.....	41
Функция <code>print()</code>	41
Функция <code>input()</code>	41
Выражения в вызовах функций	42
Завершение программы.....	42
Имена переменных.....	43
Заключение.....	44
Глава 3. Игра «Угадай число»	45
Пример запуска игры «Угадай число»	46
Исходный код игры «Угадай число»	46
Импорт модуля <code>random</code>	47
Генерация случайных чисел при помощи функции <code>random.randint()</code>	49
Приветствие игрока	51

Инструкции управления потоком.....	51
Использование циклов для повторения кода.....	51
Группировка в блоки.....	52
Создание циклов с инструкцией <code>for</code>	53
Игрок угадывает число.....	55
Преобразование значений при помощи функций <code>int()</code> , <code>float()</code> и <code>str()</code>	55
Логический тип данных.....	57
Операторы сравнения.....	58
Проверка условий и определение истинности/ложности.....	58
Эксперименты с логическими операторами, операторами сравнения и условиями.....	59
Разница между операторами <code>=</code> и <code>==</code>	61
Инструкции <code>if</code>	61
Выход из цикла до его завершения при помощи инструкции <code>break</code>	62
Проверка, победил ли игрок.....	62
Проверка, проиграл ли игрок.....	63
Заключение.....	64
Глава 4. Программа-шутник.....	65
Пример запуска программы «Шутки».....	65
Исходный код программы «Шутки».....	66
Как работает код.....	66
Экранируемые символы.....	67
Одинарные и двойные кавычки.....	68
Параметр <code>end</code> функции <code>print()</code>	70
Заключение.....	71
Глава 5. Игра «Царство драконов».....	72
Как играть в «Царство драконов».....	72
Пример запуска игры «Царство драконов».....	73
Блок-схема игры «Царство драконов».....	73
Исходный код игры «Царство драконов».....	74
Импорт модулей <code>random</code> и <code>time</code>	76
Функции в игре «Царство драконов».....	76
Инструкции <code>def</code>	76
Вызов функции.....	77
Где указывать определение функций?.....	77
Многострочный текст.....	78
Выполнение циклов с помощью инструкций <code>while</code>	79
Логические операторы.....	80
Оператор <code>and</code>	80
Оператор <code>or</code>	81
Оператор <code>not</code>	82
Вычисление логических операций.....	82
Возвращаемые значения функций.....	84
Глобальная и локальная области видимости переменных.....	84
Параметры функции.....	86
Отображение результатов игры.....	87
Определение пещеры с дружелюбным драконом.....	88

Игровой цикл	89
Вызов функций в программе.....	90
Запрос «сыграть снова»	91
Заключение.....	91
Глава 6. Использование отладчика	93
Типы багов.....	93
Отладка	95
Запуск отладчика.....	95
Пошаговое выполнение программы с помощью отладчика.....	97
Область глобальных переменных.....	97
Область локальных переменных.....	98
Обычное выполнение и завершение работы.....	98
Навигация по коду	98
Поиск багов	100
Установка точек останова.....	104
Использование точек останова	105
Заключение.....	107
Глава 7. Проектирование игры «Виселица» с помощью блок-схем	108
Правила игры «Виселица»	108
Пример запуска игры «Виселица»	109
ASCII-графика	110
Проектирование игры с помощью блок-схем	111
Создание блок-схем	112
Ветвление в блок-схемах	113
Заканчиваем или начинаем игру сначала	115
Следующая попытка.....	115
Обратная связь с игроком.....	117
Заключение.....	118
Глава 8. Написание кода игры «Виселица»	119
Исходный код игры «Виселица»	119
Импорт модуля <code>random</code>	123
Константы	123
Списки.....	124
Доступ к элементам по их индексам	125
Индекс за пределами диапазона и ошибка <code>IndexError</code>	126
Присваивание индексов элементам	126
Конкатенация списков	126
Оператор <code>in</code>	127
Вызов методов	127
Методы списков <code>reverse()</code> и <code>append()</code>	128
Строковый метод <code>split()</code>	128
Получение секретного слова из списка.....	129
Отображение игрового поля для игрока	130
Функции <code>list()</code> и <code>range()</code>	131
Срезы списков и строк.....	132
Вывод секретного слова с пробелами	134

Получение предположений игрока	136
Строковые методы <code>lower()</code> и <code>upper()</code>	136
Завершение цикла <code>while</code>	138
Инструкции <code>elif</code>	138
Проверка допустимости предположения игрока	139
Предложение игроку сыграть заново	140
Обзор функций игры	141
Игровой цикл	141
Вызов функции <code>displayBoard()</code>	142
Ввод игроком угадываемой буквы	142
Проверка наличия буквы в секретном слове	143
Проверка — не победил ли игрок	143
Обработка ошибочных предположений	144
Проверка — не проиграл ли игрок	144
Завершение или перезагрузка игры	145
Заключение	146
Глава 9. Доработка игры «Виселица»	147
Увеличение числа угадываний	147
Словари	148
Определение размера словаря с помощью функции <code>len()</code>	149
Различия между списком и словарем	150
Методы словаря <code>keys()</code> и <code>values()</code>	151
Использование словаря слов в игре «Виселица»	152
Случайный выбор из списка	152
Удаление элементов списка	154
Множественное присваивание	156
Выбор игроком категории слов	157
Заключение	158
Глава 10. Игра «Крестики-нолики»	159
Пример запуска игры «Крестики-нолики»	160
Исходный код игры «Крестики-нолики»	161
Проектирование программы	166
Данные для прорисовки игрового поля	166
Стратегия игры ИИ	167
Импорт модуля <code>random</code>	168
Вывод игрового поля на экран	169
Предоставление игроку выбора между «X» или «O»	170
Выбор — кто будет ходить первым	171
Размещение меток на игровом поле	171
Ссылки на список	172
Использование ссылок на списки в функции <code>makeMove()</code>	175
Проверка — не победил ли игрок	175
Дублирование данных игрового поля	178
Проверка — свободна ли клетка игрового поля	178
Разрешение игроку сделать ход	179
Вычисление по короткой схеме	180

Выбор хода из списка.....	182
Значение None	183
Создание искусственного интеллекта	184
Проверка — сможет ли компьютер победить, сделав ход.....	185
Проверка — сможет ли игрок победить, сделав ход.....	185
Проверка угловых, центральной и боковых клеток (в порядке очереди)	186
Проверка — заполнено ли поле.....	187
Игровой цикл	187
Выбор буквы игрока и того, кто будет ходить первым	188
Переменная <code>turn</code> в значении 'Человек'	188
Переменная <code>turn</code> в значении 'Компьютер'	190
Предложение игроку сыграть заново.....	190
Заключение.....	191
Глава 11. Дедуктивная игра «Холодно-горячо»	192
Пример запуска игры «Холодно-горячо»	193
Исходный код игры «Холодно-горячо».....	193
Блок-схема игры «Холодно-горячо»	195
Импорт модуля <code>random</code> и определение функции <code>getSecretNum()</code>	196
Перетасовка уникального набора цифр	197
Изменение порядка элементов списка с помощью функции <code>random.shuffle()</code>	197
Получение секретного числа из перетасованных цифр.....	198
Расширенные операторы присваивания.....	198
Подсчет выдаваемых подсказок.....	200
Метод списка <code>sort()</code>	201
Строковый метод <code>join()</code>	202
Проверка на содержание в строке только чисел.....	203
Начало игры	203
Интерполяция строк.....	204
Игровой цикл	205
Получение предположения игрока.....	206
Получение подсказок в зависимости от предположения игрока.....	206
Проверка победы или поражения игрока.....	207
Предложение сыграть снова.....	207
Заключение.....	208
Глава 12. Декартова система координат	209
Сетки и декартовы координаты.....	209
Отрицательные числа	211
Система координат компьютерного экрана.....	213
Математические хитрости.....	213
Хитрость 1: минус «съедает» плюс слева от себя.....	214
Хитрость 2: два минуса в сумме дают плюс	214
Хитрость 3: два слагаемых числа можно переставлять местами	214
Абсолютные величины и функция <code>abs()</code>	215
Заключение.....	216

Глава 13. Игра «Охотник за сокровищами»	217
Пример запуска игры «Охотник за сокровищами»	218
Исходный код игры «Охотник за сокровищами»	222
Проектирование программы	227
Импорт модулей <code>random</code> , <code>sys</code> и <code>math</code>	227
Создание поля для новой игры	228
Генерация игрового поля	229
Изображение координат x вдоль верхней части поля	230
Рисование океана	231
Вывод ряда в океане	232
Изображение координат x вдоль нижней части игрового поля	233
Создание случайных сундуков с сокровищами	233
Определение допустимости хода	234
Отражение хода на игровом поле	234
Поиск ближайшего сундука с сокровищами	235
Удаление значений с помощью метода списка <code>remove()</code>	238
Получение хода игрока	239
Вывод игроку инструкций по игре	241
Игровой цикл	242
Демонстрация игроку статуса игры	243
Обработка хода игрока	243
Нахождение затонувшего сундука с сокровищами	244
Проверка победы игрока	245
Проверка проигрыша игрока	245
Завершение работы программы с помощью функции <code>sys.exit()</code>	246
Заключение	246
Глава 14. Шифр Цезаря	248
Криптография и шифрование	248
Как работает шифр Цезаря	249
Пример запуска программы «Шифр Цезаря»	251
Исходный код программы «Шифр Цезаря»	252
Установление максимальной длины ключа	253
Выбор между шифрованием и расшифровыванием сообщения	254
Получение сообщения от игрока	254
Получение ключа от игрока	255
Шифрование/расшифровывание сообщения	255
Нахождение переданных строк с помощью строчного метода <code>find()</code>	256
Шифрование/расшифровка каждой буквы	257
Запуск программы	258
Полный перебор	259
Добавление режима полного перебора	260
Заключение	261
Глава 15. Игра «Реверси»	263
Как играть в «Реверси»	263
Пример запуска игры «Реверси»	266
Исходный код игры «Реверси»	268

Импорт модулей и создание констант	275
Структура данных игрового поля	275
Отображение на экране структуры данных игрового поля.....	276
Создание структуры данных нового игрового поля.....	277
Проверка допустимости хода	278
Проверка каждого из восьми направлений	279
Определение наличия фишек, которые можно перевернуть	280
Проверка допустимости координат	282
Получение списка со всеми допустимыми ходами	282
Вызов функции <code>bool()</code>	283
Получение игрового счета.....	284
Получение сделанного игроком выбора фишки.....	285
Определение первого игрока	285
Помещение фишки на поле	286
Создание копии структуры данных игрового поля.....	286
Определение того, находится ли клетка в углу	287
Получение хода игрока.....	287
Получение хода компьютера.....	290
Разработка стратегии с угловыми ходами	290
Получения списка самых результативных ходов	291
Вывод игрового счета на экран	292
Начало игры	292
Проверка на попадание в тупик	293
Выполнение хода игроком	294
Выполнение хода компьютером	295
Игровой цикл	296
Предложение сыграть снова	297
Заключение.....	298
Глава 16. Искусственный интеллект игры «Реверси»	299
Компьютер против компьютера	300
Пример запуска модели 1	300
Исходный код модели 1	301
Удаление приглашений для игрока и добавление игрока-компьютера.....	303
Компьютер против компьютера — несколько партий.....	304
Пример запуска модели 2	304
Исходный код модели 2	305
Отслеживание результатов партий	306
Отключение вызовов функции <code>print()</code>	307
Оценка искусственных интеллектов в процентном соотношении.....	307
Деление приводит к получению числа с плавающей запятой	308
Функция <code>round()</code>	309
Сравнение различных алгоритмов ИИ	309
Исходный код модели 3	310
Принципы работы искусственных интеллектов в модели 3	312
ИИ лучшего углового хода	312
ИИ худшего хода.....	312

ИИ случайного хода	313
Проверка на граничные ходы	314
ИИ лучшего углового-граничного хода	314
Сравнение искусственных интеллектов	315
ИИ худшего хода против ИИ лучшего углового хода	315
ИИ случайного хода против ИИ лучшего углового хода	316
ИИ лучшего углового-граничного хода против ИИ лучшего углового хода	317
Заключение	318
Глава 17. Создание графики	319
Установка pygame	320
Привет, pygame!	320
Пример запуска pygame-программы «Привет, мир!»	321
Исходный код pygame-программы «Привет, мир!»	322
Импорт модуля pygame	323
Инициализация pygame	324
Настройка окна pygame	324
Кортежи	325
Объекты поверхности	326
Работа с цветом	326
Вывод текста в окне pygame	327
Использование шрифтов для оформления текста	327
Рендеринг объекта Font	328
Настройка местоположения текста с помощью атрибутов Rect	329
Заливка цветом объекта Surface	331
Функции рисования pygame	332
Рисование многоугольника	332
Рисование линии	333
Рисование круга	333
Рисование эллипса	334
Рисование прямоугольника	334
Окрашивание пикселей	335
Метод blit () для объектов Surface	336
Вывод объекта Surface на экран	336
События и игровой цикл	336
Получение объектов Event	337
Выход из программы	338
Заключение	338
Глава 18. Анимированная графика	339
Пример запуска игры программы	339
Исходный код программы	339
Перемещение и контроль отскока блоков	342
Создание констант	343
Константы для направлений	344
Константы для цвета	345
Создание структуры данных блока	345

Игровой цикл	346
Обработка решения игрока завершить игру	346
Перемещение каждого блока.....	347
Управление отскакиванием блока.....	349
Отображение в окне блоков в новых положениях	350
Отображение окна на экране	350
Заключение.....	351
Глава 19. Обнаружение столкновений.....	352
Пример работы программы	353
Исходный код программы.....	353
Импорт модулей	356
Использование объекта <code>clock</code> для управления скоростью работы программы.....	357
Настройка окна и структур данных.....	357
Создание переменных для отслеживания перемещения.....	359
Обработка событий.....	359
Обработка события <code>KEYDOWN</code>	361
Обработка события <code>KEYUP</code>	363
Телепортация игрока	364
Добавление новых блоков «еды»	364
Перемещение игрока по окну	365
Отображение блока игрока в окне.....	366
Проверка на столкновения	366
Отображение блоков «еды».....	367
Заключение.....	368
Глава 20. Использование звуков и изображений	369
Добавление изображений с помощью спрайтов	369
Графические и звуковые файлы	370
Пример запуска игры.....	371
Исходный код программы.....	371
Настройка окна и создание структуры данных.....	375
Добавление спрайта.....	375
Изменение размера спрайта.....	376
Установка музыки и звуков.....	376
Добавление аудиофайлов	376
Включение/отключение звука.....	377
Отображение спрайта игрока в окне.....	378
Проверка на столкновения.....	378
Отображение спрайтов вишен в окне	379
Заключение.....	380
Глава 21. Игра «Ловкач» с графикой и звуком.....	381
Обзор основных типов данных <code>pygame</code>	381
Пример запуска игры «Ловкач»	383
Исходный код игры «Ловкач».....	383
Импорт модулей	388
Создание констант	389

Определение функций.....	390
Завершение игры и добавление паузы.....	390
Отслеживание столкновений со злодеями.....	391
Отображение текста в окне.....	392
Инициализация ругаме и настройка окна	393
Установка шрифтов, изображений и звуков.....	394
Отображение начального экрана.....	395
Начало игры.....	396
Игровой цикл	398
Обработка событий клавиатуры.....	398
Обработка событий мыши.....	400
Добавление новых злодеев	401
Перемещение спрайтов игрока и злодеев.....	402
Реализация чит-кодов	404
Удаление спрайтов злодеев.....	404
Отображение окна.....	405
Отображение очков игрока	405
Отображение спрайтов игрока и злодеев	406
Проверка на столкновения.....	406
Экран окончания игры.....	407
Изменение игры «Ловкач»	408
Заключение.....	409
Предметный указатель	410

ОБ АВТОРЕ

Эл Свейгарт — разработчик программного обеспечения, автор технических книг и человек, у которого всегда при себе полотенце*. Он написал несколько книг по программированию для начинающих, в том числе «Программирование для детей. Делай игры и учи язык Scratch!». Его книги на языке оригинала можно бесплатно прочитать на сайте www.inventwithpython.com.

* В оригинале: «...who really knows where his towel is» — отсылка к роману «Автостопом по галактике» (1979 г.) Дугласа Адамса, британского писателя, оказавшего влияние на культовое для программистов на Python комедийное шоу «Летающий цирк Монти Пайтона» (Monty Python's Flying Circus). — *Примеч. ред.*

ВВЕДЕНИЕ



Когда я в детстве впервые запустил компьютерную игру, меня моментально затянуло. Но я хотел не только играть в игры, я хотел их создавать. Я нашел книгу, похожую на эту, которая научила меня писать свои первые программы и игры.

Это было просто и весело. Первые игры, которые я делал, были простыми и похожими на показанные в этой книге. Они не были такими вычурными, как бестселлеры Nintendo, которые мне покупали родители, но это были игры, которые я сделал сам.

Сейчас, став совершеннолетним, я все еще получаю удовольствие от программирования и мне за это еще и платят. Но даже если вы не хотите связывать с этим жизнь, программирование — полезный и веселый навык. Он тренирует ваш мозг мыслить логически, строить планы и пересматривать свои решения каждый раз, как вы обнаружите ошибку в коде.

Многие книги, посвященные программированию для начинающих, попадают под две категории. Первая категория включает книги, которые обучают не столь программированию, сколько «программному обеспечению для создания игр» или языкам, которые упрощаются настолько, что предмет изучения уже и нельзя назвать программированием. Вторая категория состоит из книг, которые обучают программированию подобно математическому учебнику — со всеми принципами и понятиями и парочкой реальных приложений в пример читателю. Эта книга прибегает к иному подходу и учит вас, как программировать с помощью создания видеоигр. Я показываю исходный код для игр и напротив объясняю принцип программирования из примера. Такой подход был ключевым для меня, когда я учился программированию.

Чем больше я узнавал, как работают программы, созданные другими людьми, тем больше идей ко мне приходило касательно моих собственных.

Все, что вам нужно — компьютер, бесплатное программное обеспечение, которое называется интерпретатор Python и эта книга. Как только вы узнаете, как создавать игры из этой книги, вы сможете разрабатывать игры сами.

Компьютеры — поразительные механизмы и обучение программированию не такое неподъемное, как сначала кажется. Компьютерная *программа* — это набор инструкций, которые компьютер может понять, точно так же, как книга с историями — это куча предложений, которые может понять читатель. Чтобы создать инструкцию, вы пишете программу на языке, который компьютер понимает. Эта книга научит вас языку программирования Python. Существует множество других языков программирования, которые вы можете освоить, например, BASIC, Java, JavaScript, PHP и C++.

В детстве я выучил BASIC, но более новые языки программирования типа Python учатся гораздо легче. Python также используется профессиональными программистами в работе и любителями в качестве хобби. Вдобавок он абсолютно бесплатный для установки и использования — вам всего-то нужен Интернет, чтобы скачать его.

Поскольку видеоигры — ничто иное как компьютерные программы, они также состоят из инструкций. Игры, которые вы создадите благодаря этой книге, будут проще сравнительно с играми для Xbox, PlayStation или Nintendo. В этих играх не будет невероятной графики, потому что они рассчитаны на то, чтобы обучить вас основам кодирования. Они преднамеренно просты, поэтому вы можете сосредоточиться на обучении программированию. Игры не должны быть сложными, чтобы быть интересными!

Для кого эта книга?

Программирование — это не сложно, сложнее найти материалы, которые обучат вас делать интересные вещи с его помощью. Другие книги предоставляют слишком много того, что начинающему кодеру не понадобится. Эта книга научит вас программировать собственные игры; вы получите полезный опыт и интересные видеоигры, которые сможете продемонстрировать.

Эта книга для:

- абсолютных новичков, которые хотят обучиться программированию, даже не имея при этом предыдущего опыта;
- детей и подростков, которые хотят научиться программированию, создавая игры;
- взрослых и учителей, которые хотят освоить программирование;
- кого угодно, от мала до велика, кто хочет научиться программировать, изучая профессиональный язык программирования.

О книге

В большинстве разделов этой книги вводится и объясняется по одному новому игровому проекту. В некоторых главах рассматриваются дополнительные полезные темы, такие как отладка. Новые концепции программирования по мере применения их в играх и главы предназначены для чтения по порядку.

Вот краткое изложение того, что вы найдете в каждой главе:

- Глава 1 «Интерактивная среда разработки» пошагово объясняет, как интерактивная среда разработки Python может использоваться для экспериментов с кодом.
- Глава 2 «Написание программ» объясняет, как писать полноценные программы в редакторе файлов Python.
- В главе 3 «Игра “Угадай число”» вы запрограммируете первую игру в этой книге, в которой попросите игрока угадать секретное число, а затем дадите подсказки относительно того, насколько его предположение близко или далеко от правильного ответа.
- В главе 4 «Программа-шутник» вы напишете простую программу, которая рассказывает пользователю несколько шуток.
- В главе 5 «Игра “Царство драконов”» вы запрограммируете игру для угадывания, в которой игрок должен выбрать между двумя пещерами: в одной живет дружелюбный дракон, а в другой — голодный.
- Глава 6 «Использование отладчика» описывает способы использования отладчика для устранения неполадок в коде.
- Глава 7 «Проектирование игры “Виселица” с помощью блок-схем» объясняет, как блок-схемы помогают в планировании более длинных программ, таких как игра «Виселица».
- В главе 8 «Написание кода игры “Виселица”» вы напишете игру «Виселица», опираясь на блок-схему из главы 7.
- Глава 9 «Доработка игры “Виселица”» расширяет игру «Виселица» новыми функциями, используя базу данных словаря Python.
- В главе 10 «Игра “Крестики-нолики”» вы узнаете, как писать игру типа «человек против компьютера», использующую искусственный интеллект.
- В главе 11 «Дедуктивная игра “Холодно-горячо”» вы узнаете, как сделать дедуктивную игру, в которой игрок должен угадать секретные числа на основе подсказок.
- Глава 12 «Декартова система координат» объясняет декартову систему координат, которую вы позже будете использовать в играх.
- В главе 13 «Игра “Охотник за сокровищами”» вы узнаете, как написать игру, в которой игрок прочесывает океан в поисках потерянных сундуков с сокровищами.