

# ПрОГРАММИРОВАНИЕ

## для детей

[Купить книгу на сайте kniga.biz.ua >>>](http://kniga.biz.ua)

# Содержание

8	<b>ВВЕДЕНИЕ</b>
10	<b>КАК УСТРОЕНА ЭТА КНИГА</b>

## 1 ЧТО ТАКОЕ ПРОГРАММИРОВАНИЕ?

14	Что такое компьютерная программа?	64	Решения и ветвление
16	Думай как компьютер	66	Считывание и распознавание
18	Как стать программистом	68	Сложные циклы
		70	Обмен сообщениями
		72	Создание блоков
		74	<b>Проект 3: Бешеные обезьяны</b>
		82	Приступим к экспериментам!

## 2 НАЧНЕМ СО SCRATCH

22	Что такое Scratch?	3	<b>ИГРЫ С PYTHON</b>
24	Установка и запуск Scratch	86	Что такое Python?
26	Интерфейс Scratch	88	Установка Python
28	Спрайты	92	Знакомство с IDLE
30	Цветные блоки и скрипты	94	Ошибки
32	<b>Проект 1: Убеги от дракона!</b>	96	<b>Проект 4: Дом с привидениями</b>
38	Перемещение объектов	98	Разбор «Дома с привидениями»
40	Костюмы	100	Выполнение программ
42	Прятки	102	Простые команды
44	События	104	Команды посложнее
46	Простые циклы	106	Разные окна
48	Перья и черепашки	108	Переменные в Python
50	Переменные	110	Типы данных
52	Вычисления	112	Вычисления в Python
54	Строки и списки	114	Строки в Python
56	Координаты	116	Ввод и вывод
58	Пошумим!	118	Принятие решений
60	<b>Проект 2: Катись, кубик</b>	120	Ветвление
62	Истина или ложь?		

122	Циклы в Python
124	Цикл while
126	Выход из цикла
128	Списки
130	Функции
132	<b>Проект 5: Забавные фразы</b>
134	Кортежи и словари
136	Списки в переменных
138	Переменные и функции
140	<b>Проект 6: Чертежный автомат</b>
148	Ошибки и отладка
150	Алгоритмы
152	Библиотеки
154	Создание окон
156	Цвета и координаты
158	Рисование фигур
160	Изменение рисунков
162	Реакция на события
164	<b>Проект 7: Охотник за пузырями</b>
176	Что дальше?

## Ч УСТРОЙСТВО КОМПЬЮТЕРОВ

180	Внутри компьютера
182	Двоичная система
184	Символы и коды
186	Логические вентили

188	Процессоры и память
190	Необходимые программы
192	Хранение данных в файлах
194	Интернет

## 5 ПРОГРАММИРОВАНИЕ В РЕАЛЬНОМ МИРЕ

198	Компьютерные языки
200	Звезды программирования
202	Трудолюбивые программы
204	Компьютерные игры
206	Мобильные приложения
208	Программирование для интернета
210	Использование JavaScript
212	Зловредные программы
214	Мини-компьютеры
216	Стань знатоком программирования
218	<b>ГЛОССАРИЙ</b>
220	<b>АЛФАВИТНЫЙ УКАЗАТЕЛЬ</b>
224	<b>БЛАГОДАРНОСТИ</b>

# Введение

Совсем недавно компьютерное программирование казалось таинственным ремеслом, уделом специалистов. Мысль о том, что программирование может быть увлекательным занятием для каждого, большинству людей и в голову не приходила. Но мир изменился. Интернет, электронная почта, социальные сети, смартфоны и мобильные приложения, ураганом влетев в нашу жизнь, преобразили ее всего за несколько лет.

Компьютеры занимают очень важное место в современном мире, и мы принимаем это как должное. Вместо звонков по телефону мы посылаем текстовые сообщения или используем социальные сети. Мы охотно поглощаем любые плоды компьютеризации — от интернет-шопинга и развлечений до новостей и игр. Однако мы можем не только использовать эти технологии — научившись программировать, мы можем развивать их, создавать собственные произведения цифрового искусства.

Всем, что делает компьютер, управляют строки программного кода, введенные с клавиатуры. Компьютерный код похож на иностранный язык, но язык этот может освоить каждый и довольно быстро. Многие считают, что программирование — одно из наиболее важных в XX веке умений.

Учиться программировать очень интересно, ведь ты можешь получить результаты сразу же, и не важно, сколько еще материала предстоит изучить. Более того, создание игр и программ — такое увлекательное занятие, что очень скоро покажется, будто оно почти не требует усилий. Это отличная возможность для творчества, наверное, первая область науки, совмещающая искусство, логику, сочинительство и бизнес.

Кроме того, умение программировать очень пригодится в жизни. Оно развивает логику и интеллект, которые важны в самых разных областях — от науки и инженерного дела до медицины и юриспруденции. Количество вакансий, где нужно умение программировать, будет со временем только расти, причем хороших программистов не хватает уже сейчас. Научись программированию — и цифровой мир будет открыт для тебя!

A handwritten signature in black ink, appearing to read "Carol Vorderman". The signature is fluid and cursive, with a long horizontal line extending from the end of the last name.

КЭРОЛ ВОРДЕРМАН

# Как устроена эта книга

Эта книга знакомит с основными понятиями, которые нужно усвоить, чтобы научиться программировать. Еще ты найдешь здесь увлекательные проекты, воплощающие идеи программирования на практике. Чтобы информацию было легче понять и усвоить, она разбита на небольшие блоки.

Каждая тема рассматривается подробно, с примерами и упражнениями

42 НАЧНЕМ СО SCRATCH

## Прятки

Добро пожаловать в студию спецэффектов! Фиолетовые блоки Looks («Внешность») могут научить спрайты исчезать и появляться, увеличиваться и уменьшаться, таять и превращаться.

**Прячем спрайты**

Если нужно, чтобы спрайт исчез, используй блок `hide` («скрыть»). Спрайт останется на сцене и сможет по ней двигаться, но будет незаметен до тех пор, пока блок `«показать»` не вернет ему видимость.

Цветные рисунки поясняют различные понятия программирования

Программный код и скрипты объясняются построчно

Инструкции указывают, что надо нажать, перетащить или выделить

В блоках «Смотри также» перечислены другие относящиеся к теме разделы

СМОТРИ ТАКЖЕ  
38-39 Перемещение объектов  
Обмен сообщениями 70-71

Используй блок `hide` («скрыть»), если нужно, чтобы спрайт в игре исчез

СОВЕТЫ ЭКСПЕРТА  
Показать спрайт

Этот блок прячет кота

Этот блок разворачивает кота по часовой стрелке

Кот все еще движется, хоть и невидим

Этот блок снова показывает кота

Показать скрытый спрайт

Sprite1  
x: 84 y: -69  
rotation style: can drag in player:   
show:

Пиксельные человечки дают подсказки и советы.

170 ИГРЫ С ПУЗЫРЯМИ

## ОХОТНИК ЗА ПУЗЫРЯМИ

Вычисляем расстояние между

В этой игре, как и во многих других, нужно вычислить расстояние между двумя объектами. Для расчета используется известная математическая формула

11 Эта функция вычисляет расстояние между двумя объектами. Добавь этот код сразу после кода, введенного на шаге 9.

```
from math import sqrt
def distance(id1, id2):
    x1, y1 = get_coords(id1)
    x2, y2 = get_coords(id2)
    return sqrt((x2 - x1)**2 + (y2 - y1)**2)
```

## Размеры и эффекты

В скриптах можно менять размер спрайта и добавлять ему спецэффекты.

Выбери спецификацию из выпадающего меню.

Эффект `rickle` («всплеск») делает спрайт размытым.

change `size`

Вводи положительные числа, чтобы увеличить спрайт, или отрицательные, чтобы его уменьшить.

set `size` to

Большие числа увеличивают спрайт, а маленькие уменьшают. Число 100 — это его обычный размер.

clear graphic effects

△ Изменение размера спрайта

Эти два блока могут увеличивать или уменьшать спрайт — на определенную величину либо в процентах от его размера.

△ Добавление эффектов

Графические эффекты изменяют вид спрайта, искажая его, делая интересно искаженным.

set `color`

clear graphic effects

## Телепортация с помощью эффектов

Загрузи из библиотечной категории Fantasy («Фантастика») спрайт привидения и добавь ему этот скрипт. Привидение телепортируется, если кликнуть по нему мышкой.

when this sprite clicked

clear graphic effects

repeat (20)

change `ghost` effect by

glide  secs to x:  y:

repeat (20)

change `ghost` effect by

Этот блок секции `«Операторы»` вы случайно позиционируете по горизонтали

Этот блок `ghost` («привидение») делает спрайт чуть прозрачнее; после 20-кратного повторения этого эффекта спрайт полностью исчезнет

Этот блок медленно перемещает невидимый спрайт

Подписи поясняют каждый шаг

ЗЫРЯМИ

## Соединение между точками

других, полезно знать расстояние и расчета расстояний мы воспользуемся формулой.

Вывод:  $\sqrt{x_2^2 + y_2^2}$

Загружает функцию `sqrt` из модуля `Math`

Получает координаты первого объекта

Получает координаты второго объекта

Возвращает расстояние между объектами

ПРЯТКИ 43

Берет спецэффект, задающий меню. Эффект `pixelate` (укрупнение пикселей) делает спрайт размытым.

change `pixelate` effect by 25

et `[color v]` effect to 0

clear graphic effects

Добавления спецэффектов

Графические эффекты служат для изменения вида спрайта или изменения его формы. С ними интересно экспериментировать.

Никто не знает, где я появлюсь в следующий раз!

спрайт) делает спрайт 20-кратного размера

от блок секции Operators (операторы) выбирает случайную позицию горизонтально

`pick random (-150) to (150)`

Этот блок выбирает случайную позицию по вертикали

Этот блок позволяет спрайту снова появиться

Семь проектов для развития навыков программирования. Страницы проектов выделены синей полосой

Простые пошаговые инструкции проведут тебя по каждому из проектов

ОХОТНИК ЗА ПУЗЫРЯМИ

171

13

Дополни главный игровой цикл вызовами только что созданных функций. Убедись, что все части кода стоят на своих местах, их порядок важен. Запусти программу — пузыри должны лопаться при столкновении с подводкой. Счет игры печатается в окне консоли.

```
score = 0
#MAIN GAME LOOP
while True:
    if randint(1, BUB_CHANCE) == 1:
        create_bubble()
    move_bubbles()
    clean_up_bubs()
    score += collision()
    print(score)
    window.update()
    sleep(0.01)
```

В начале игры устанавливается количество очков в 0

Создает новые пузыри

Прибавляет очки за пузыри к общему счету

Печатает счет в окне консоли — позже мы будем выводить его как положено, в окне игры

Приостанавливает игру на короткий промежуток времени — попробуй убрать эту строку и посмотри, что произойдет

## СОВЕТЫ ЭКСПЕРТА

## Сокращенная запись в Python

Код `score += collision()` — это короткая форма записи кода `score = score + collision()`, который складывает очки за столкновение и общий счет и заносит результат в общий счет. Это типичная программная конструкция, поэтому для нее и есть короткая форма записи. То же можно делать с вычитанием, например `score -= 10` аналогично `score = score - 10`.

Не забудь сохранить свою работу.

Пометки с указанием на строку, чтобы ничего не перепутать

Эта иконка указывает, что продолжение проекта на следующей странице

В рамочках — дополнительная информация: подсказки, определения, важные для запоминания вещи

Читай и начинай программировать!

## СОВЕТЫ ЭКСПЕРТА

## Сохранение

Иконка сохранения появляется на разворотах с проектами. Она напоминает тебе сохранять выполненную работу, чтобы ничего не потерялось в случае проблем с компьютером.

Сохраняй работу почаще.



[Купить книгу на сайте kniga.biz.ua >>>](#)

# Что такое программирование?

# Что такое компьютерная программа?

Компьютерная программа — это набор инструкций, следуя которым компьютер выполняет поставленную задачу. Программировать — значит писать для компьютера пошаговые инструкции, объясняющие, что и как ему нужно делать.

## Компьютерные программы — повсюду

Нас окружают компьютерные программы. Множеством приборов и мобильных устройств, которыми мы пользуемся изо дня в день, управляют программы. То есть эти устройства следуют пошаговым инструкциям, которые написаны программистами.

### СМОТРИ ТАКЖЕ

Думай как компьютер [16-17 >](#)

Как стать программистом [18-19 >](#)

#### △ Мобильные телефоны

Программы позволяют сделать звонок или отправить текстовое сообщение. При поиске контакта по имени программа сама находит нужный телефонный номер.

#### △ Компьютерные приложения

Все, что мы делаем с помощью компьютера, от просмотра сети интернет до составления документов и проигрывания музыки, возможно благодаря коду, написанному программистами.

#### △ Игры

Игровая приставка — это всего лишь специальный тип компьютера. Все игры, сделанные для приставок, — это программы. Графика, звук и управление в играх расписаны в программном коде.

#### ▷ Автомобили

В некоторых автомобилях компьютерные программы следят за скоростью, температурой воздуха и количеством топлива в баке.

Программы могут даже помочь вовремя притормозить, чтобы поездка была безопаснее.

#### △ Стиральные машины

Стиральные машины программируются на разные режимы стирки. Компьютерный код следит за температурой воды и временем стирки.

## Как работают компьютерные программы

Компьютеры могут казаться очень умными, но это всего лишь напичканные электроникой ящики, которые умеют очень быстро и точно выполнять инструкции. Мы, разумные существа, можем давать компьютерам задачи, описывая их в виде программ, то есть пошаговых инструкций.

### 1 Компьютеры не умеют думать

Сам по себе компьютер ни на что не способен. Задача программиста — давать ему инструкции.

Без инструкций  
компьютер совсем  
бестолковый

Это компьютерная программа,  
выполняющая обратный отсчет  
перед запуском

### 2 Напиши программу

Ты можешь объяснить компьютеру, что делать, написав очень подробные инструкции, которые называются программой. Каждая инструкция должна быть достаточно простой, чтобы компьютер ее понял. Если инструкция написана неверно, компьютер поведет себя не так, как ты хочешь.

```
for count in range(10, 0, -1):  
    print("Counting down", count)
```

### 3 Языки программирования

Компьютер может выполнять инструкции лишь на тех языках, которые ему понятны. Программистам приходится выбирать, какой язык лучше подходит для решения конкретной задачи.

```
for count in range(10, 0, -1):  
    print("Counting down", count)
```

В итоге все программы становятся двоичным кодом — простейшим компьютерным языком, состоящим из нулей и единиц

СЛЕНГ

## Софти железо

Железо (или аппаратные средства) — это части компьютера, которые ты можешь увидеть или потрогать (проводы, схемы, клавиатура, экран и так далее). Софтом, или программными средствами, называют программы, которые выполняются на компьютере и управляют его работой. Аппаратные и программные средства работают сообща, чтобы компьютер мог выполнять полезные задачи.

# Пуск!

# Думай как компьютер

Программист должен научиться думать как компьютер. Все задачи нужно разбивать на небольшие подзадачи, которым легко следовать и которые невозможно понять неправильно.

## Думай как робот

Представь себе кафе с официантом-роботом. Поскольку у робота бесхитростный компьютерный мозг, ему нужно объяснить, как доставлять блюда с кухни посетителям, ожидающим в обеденном зале. Сперва эту проблему нужно разбить на простые, понятные компьютеру подзадачи.

### 1 Программа 1 для робота-официанта

Следуя этой программе, робот берет еду с тарелки, вламывается из кухни в обеденный зал прямо сквозь стену и кладет еду на пол. Этот алгоритм составлен недостаточно подробно.

1. Взять еду
2. Идти из кухни к столу посетителя
3. Положить еду

### 2 Программа 2 для робота-официанта

На этот раз мы сказали роботу, чтобы он шел через кухонную дверь. Так он и делает, но затем натыкается на местного кота, падает и разбивает тарелку.

1. Взять тарелку с едой
2. Идти из кухни к столу посетителя таким образом:
  - идти к кухонной двери
  - идти от двери к столу посетителя
3. Поставить тарелку на стол перед посетителем

## СМОТРИ ТАКЖЕ

◀ 14–15 Что такое компьютерная программа?

Как стать 18–19 ▶  
программистом

## СЛЕНГ

### Алгоритм

Алгоритм — это набор простых инструкций, объясняющих, как выполнить задачу. Программа — это алгоритм, переведенный на понятный компьютеру язык.

### ▷ Беда!

Инструкции слишком неоднозначны: мы забыли объяснить роботу, что идти нужно через дверь. Людям это показалось бы очевидным, но компьютеры не умеют думать сами.

### △ Все еще не идеально

Робот не знает, как быть с препятствиями (вроде кошек). Чтобы он мог безопасно перемещаться, программа должна содержать еще более подробные инструкции.

### 3 Программа 3 для робота-официанта

Следуя этому варианту программы, робот благополучно доставляет блюдо посетителю, огибая препятствия. Однако, поставив тарелку на стол, робот застывает на месте, а на кухне тем временем скапливается приготовленная еда.

1. Взять тарелку с едой и держать ее ровно
2. Идти из кухни к столу посетителя таким образом:

идти к кухонной двери

если замечено препятствие, обойти его

идти от двери к столу посетителя

если замечено препятствие, обойти его

3. Поставить тарелку на стол перед посетителем

#### △ Неужели получилось?

Наконец-то робот смог благополучно доставить еду. Но мы забыли сказать ему, что потом нужно вернуться на кухню за новой тарелкой.

## Пример из жизни

Может, наш робот-официант и выдуманный, однако алгоритмы такого типа встречаются повсюду. Например, управляемый компьютером лифт имеет дело со схожими проблемами. Вверх или вниз ему следует двигаться? На какой следующий этаж ехать?

1. Ждать до закрытия дверей

2. Ждать нажатия кнопки

Если нажата кнопка этажа выше текущего:

начать движение вверх

Если нажата кнопка этажа ниже текущего:

начать движение вниз

3. Ждать, пока этаж не будет соответствовать нажатой кнопке

4. Открыть двери

#### △ Программа для лифта

Чтобы лифт работал правильно и безопасно, каждый шаг программы должен быть точным, понятным и учитывать все возможные случаи. Программисты должны убедиться, что они составили подходящий алгоритм.

# Как стать программистом

Программисты (или кодеры) — это люди, которые создают программы, управляющие всем, что мы видим и делаем на компьютере. Ты сможешь писать собственные программы, если освоишь какой-нибудь язык программирования.

## Языки программирования

Языков программирования очень много. Каждый из этих языков можно использовать для выполнения разных задач. Вот некоторые из самых популярных языков и цели, для которых их чаще всего используют.

**C**

Мощный язык для написания компьютерных операционных систем.

**Ada**

Используется для управления космическими кораблями, спутниками и самолетами.

**Java**

Работает на компьютерах, мобильных телефонах и планшетах.

**Scratch**

Визуальный язык, который идеально подходит для изучения программирования.

**MATLAB**

Идеален, когда нужно выполнять много математических вычислений.

**Ruby**

Для автоматического отображения информации на интернет-страницах.

**Javascript**

Язык для написания интерактивных веб-сайтов.

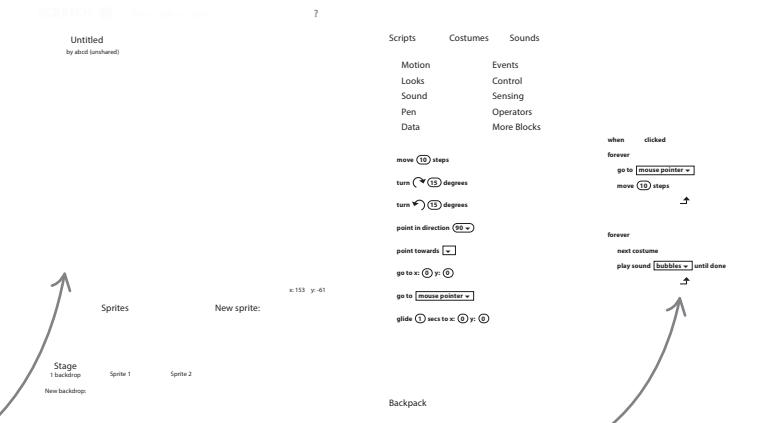
**Python**

Текстовый язык, подходящий для самых разных целей.

## Что такое Scratch?

Scratch (читается «скрэтч») отлично подходит, чтобы начать программировать. Программы на этом языке состоят не из текстовых инструкций, а из блоков кода, соединенных между собой. Scratch — простой и эффективный язык, знакомящий с основными понятиями, необходимыми, чтобы программировать на других языках.

Программа запускается на этой части экрана



Код создают, соединяя цветные блоки между собой

## СМОТРИ ТАКЖЕ

Что такое Scratch? **22–23 >**

Что такое Python? **86–87 >**

## Что такое Python?

Люди по всему миру используют Python (читается «пайтон») для создания игр, приложений и веб-сайтов. Освоив этот замечательный язык, можно писать самые разные программы. Код на Python состоит из слов английского языка и символов.

Программа, написанная на Python

```
ghostgame
# Ghost Game
from random import randint
print('Ghost Game')
feeling_brave = True
score = 0
while feeling_brave:
    ghost_door = randint(1, 3)
    print('Three doors ahead...')
```

## Приступим

Пора начать программировать! Все, что тебе нужно, — компьютер с подключением к интернету. В этой книге мы начнем с языка Scratch, который отлично подходит, чтобы встать на путь программиста-знатока. Приготовься войти в чудесный мир компьютерного программирования!

### СОВЕТЫ ЭКСПЕРТА

#### Экспериментируй!

Тебе как программисту-новичку полезно экспериментировать с кодом своих программ. Один из лучших способов чему-то научиться — это менять разные части кода и смотреть, что произойдет. Играя с программой, ты найдешь новые способы решения задач, узнаешь больше о программировании, и тебе станет еще интереснее!