

ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ IT-КУБ ГОРОДА КОСТРОМЫ



УТВЕРЖДАЮ  
Директор Гимназии №33  
города Костромы  
Меркурьева Н.В.

2026 г.

Дополнительное образование

УЧЕБНАЯ ПРОГРАММА ПО НАПРАВЛЕНИЮ

---

**Программирование на Python**

---

**5 класс**

**Составитель: Данченкова Александра Геннадьевна**

Программа рассмотрена и одобрена на совещании ПМК  
Протокол №5 от «12» января 2026 г.

## 1. Пояснительная записка

Программа разработана на основе:

— Федерального Закона РФ «Об образовании в Российской Федерации» от 29.12.2012 года № 273-ФЗ;

— Федерального государственного образовательного стандарта основного общего образования (приказ от 17 декабря 2010 года № 1897);

— Примерной основной образовательной программы основного общего образования, одобренной решением федерального учебно-методического объединения по общему образованию (протокол от 08.04.2015 г. № 1/15);

— Примерной основной образовательной программы среднего общего образования, одобренной решением федерального учебно-методического объединения по общему образованию (протокол от 28 июня 2016 г. № 2/16-з);

— Концепции развития дополнительного образования, утвержденной распоряжением Правительства Российской Федерации от 4 сентября 2014 г. № 1726-р;

— Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам, утвержденным приказом Министерства просвещения Российской Федерации № 196 от 09.11.2018г;

— Постановления Главного государственного санитарного врача РФ от 04.07.2014 № 41 «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей»;

— Учебного плана Центра цифрового образования «IT-куб» в Костромской области на базе МБОУ «Гимназия № 33» города Костромы.

**Актуальность** программы отвечает современным потребностям детей и родителей в получении необходимых теоретических знаний и навыков в сфере программирования для личностного развития и успешной адаптации к требованиям жизни в современном информационном обществе.

Методы, применяемые в процессе обучения, такие как проблемное обучение, проектная деятельность, способствуют формированию мотивации учащихся к углубленному изучению программирования, как одной из компьютерных наук. У обучающихся формируется познавательный интерес, самостоятельность мышления, стремление к самопознанию.

Отличительная особенность программы состоит в том, что она позволяет привлечь обучающихся среднего школьного возраста к изучению программирования при помощи языка Python, так как он обладает следующими достоинствами:

- Python – это текстовый язык программирования. Он универсален, пригоден для создания самых разных программ, от текстовых процессоров до веб-браузеров;

- Python – простой и удобный язык. По сравнению со многими другими языками читать и составлять программы на Python совсем не сложно;

- В Python есть библиотеки готовых процедур для использования в своих программах. Это позволяет создавать сложные программы быстро;

- Python используется как язык программирования крупными корпорациями, например, такими как Google.

**Аудитория:** обучающиеся 5 класса.

**Состав группы:** 10-12 человек.

**Форма обучения:** очная.

**Объем и срок освоения программы:** срок реализации программы - 1 год. Общая продолжительность образовательного процесса составляет 68 часов.

**Режим:** Занятия проводятся 2 раза в неделю по 1 часу.

**Цель программы:** способствовать развитию алгоритмического и логического мышления посредством языка программирования Python.

## **Задачи программы:**

### *— Обучающие:*

- сформировать представление об основных элементах программирования;
- познакомить с синтаксисом языка программирования Python;
- сформировать навыки работы в интегрированной среде разработки на языке Python;
- способствовать приобретению навыков разработки эффективных алгоритмов и программ на основе изучения языка программирования Python.

### *— Развивающие:*

- совершенствовать аналитические навыки;
- способствовать формированию алгоритмического и логического мышления;
- совершенствовать навык поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- развивать умение планировать свои действия с учётом фактора времени.

### *— Воспитательные:*

- воспитывать в обучающихся усидчивость, аккуратность, умение доводить начатое дело до конца;
- формировать коммуникативные навыки.

## **2. Планируемые результаты освоения программы:**

### *Предметные результаты:*

- иметь представление об основных классических алгоритмах и способах их реализации;
- знать синтаксис языка программирования Python;
- иметь представление о величине и ее характеристиках;
- знать простые и сложные структуры данных, а также конструкции для работы с ними;
- иметь представление о некоторых модулях (turtle, random, tkinter и др.);
- уметь анализировать как свой, так и чужой код;
- иметь представление о функциях, владеть навыками работы с ними.

### *Метапредметные результаты:*

- учащиеся способны осуществлять рефлексивную деятельность, оценивать собственные результаты, корректировать дальнейшую деятельность по программированию;
- сформировано алгоритмическое и логическое мышление;
- обладают навыками поиска информации в сети Интернет, анализа выбранной информации на соответствие запросу, использования информации при решении задач;
- сформировано умение планировать деятельность с учетом фактора времени.

### *Личностные результаты:*

- учащиеся проявляют усидчивость, аккуратность, умеют доводить начатое дело до конца;
- сформировано умение работать в коллективе.

## **3. Формы и виды учебной деятельности**

### *Формы организации учебных занятий:*

- *фронтальная* – подача материала всей учебной группе учащихся;
- *индивидуальная* – самостоятельная работа учащихся с оказанием педагогом помощи при возникновении затруднения;
- *групповая* – предоставление учащимся возможности самостоятельно построить свою деятельность, ощутить помощь со стороны друг друга, учесть возможности каждого на конкретном этапе деятельности.

### *Формы проведения занятий:*

- *вводное занятие* – педагог знакомит учащихся с техникой безопасности, особенностями организации деятельности и предлагаемым планом работы на текущий год;
- *ознакомительное занятие* – педагог знакомит учащихся с новыми методами работы в зависимости от темы занятия;

- *тематическое занятие* – обучающимся предлагается работать над моделированием по определенной теме. Занятие содействует развитию творческого воображения учащихся;

- *занятие-проект* – на занятии учащиеся получают полную свободу в выборе направления работы, не ограниченного определенной тематикой. Учащиеся, участвующие в работе по выполнению предложенного задания, рассказывают о выполненной работе, о ходе выполнения задания, о назначении выполненного проекта;

- *комбинированное занятие* – проводится для решения нескольких учебных задач;

- *итоговое занятие* – служит подведению итогов работы за учебный год, может проходить в виде мини-выставок, просмотров творческих работ и презентаций.

#### 4. Формы контроля результатов освоения программы.

Формы аттестации: наблюдение, опрос, защита проекта, самостоятельная работа, конкурс работ.

Итоговый контроль проводится в конце года с целью определения степени достижения результатов обучения и получения сведений для совершенствования программы и методов обучения.

#### 5. Учебный план

№ п/п	Название разделов, тем	Количество часов			Форма аттестации/ контроля
		Всего	Теория	Практика	
<b>1</b>	<b>Введение в язык программирования Python</b>	<b>30</b>	<b>12</b>	<b>18</b>	
1.1	Устройство языка Python. Среда разработки IDE.	2	1	1	Наблюдение
1.2	Вывод данных на языке Python. Ввод и отладка программ в среде	2	1	1	Наблюдение
1.3	Переменные	2	1	1	Опрос
1.4	Выражения. Типы данных	3	1	2	Опрос
1.5	Строки	5	2	3	Опрос
1.6	Списки	5	2	3	Опрос
1.7	Кортежи и словари	5	2	3	Опрос
1.8	Модуль turtle	6	2	4	Конкурс работ
<b>2</b>	<b>Основы языка Python</b>	<b>38</b>	<b>10</b>	<b>28</b>	
2.1	Условный оператор	3	1	2	Опрос
2.2	Объединение условий	3	1	2	Самостоятельная работа
2.3	Использование цикла for	4	1	3	Опрос
2.4	Использование цикла while	4	1	3	Опрос
2.5	Решение задач на циклические алгоритмы	4	1	3	Самостоятельная работа
2.6	Применение функций	4	1	3	Самостоятельная работа
2.7	Использование встроенных функций	4	1	3	Опрос
2.8	Графика с модулем tkinter	4	1	3	Самостоятельная работа
<b>3</b>	<b>Итоговый проект</b>	<b>8</b>	<b>2</b>	<b>6</b>	Защита проекта
	<b>Итого:</b>	<b>68</b>	<b>22</b>	<b>46</b>	

## **6. Содержание учебного плана**

### **Раздел 1. Введение в язык программирования Python**

#### **Тема 1.1. Устройство языка Python. Среда разработки IDE. Сохранение Python-программ**

Теория. Техника безопасности на занятии. Понятие «алгоритм», «исполнитель», «язык программирования», «программа», «интерпретатор». История языка программирования Python и его возможности. Виды окон в IDE: окно программы и окно консоли. Сравнение этих окон и их возможностей.

Практика. Сохранение и запуск python-программ в среде разработки IDE.

#### **Тема 1.2. Вывод данных на языке Python. Ввод и отладка программ в среде**

Теория. Правила синтаксиса Python: правило начала, правило порядка, правило регистра. Понятие функции. Функция print().

Практика. Проект «Символьная графика». Создание определённого рисунка с помощью символов. Отработка функции print().

#### **Тема 1.3. Переменные**

Теория. Понятие «переменная». Правила именования переменных в языке Python. Оператор присваивания. Функция input().

Практика. Проект «Аватар». В данном проекте обрабатывается функция input(), с помощью которой становится возможным ввести свои данные в программу и отобразить их.

#### **Тема 1.4. Выражения. Типы данных**

Теория. Арифметические операции с помощью математических операторов +, -, \*, /. Порядок выполнения операций. Понятие «выражение», «типы данных». Функции int() и str().

Практика. Проект «Сумматор». При написании данной программы обрабатываются математические операторы и функции int() и str().

#### **Тема 1.5. Строки**

Теория. Понятие «строка». Создание строк. Переменные внутри строк. Операции со строками.

Практика. Решение задач на обработку операций со строками.

#### **Тема 1.6. Списки**

Понятие «список». Создание списков. Добавление/удаление элементов в/из список/списка. Операции со списками.

Практика. Решение задач на обработку операций со списками.

#### **Тема 1.7. Кортежи и словари**

Теория. Понятие «кортеж». Создание кортежа. Операции с кортежем. Понятие «словарь». Создание словаря.

Практика. Проект «Любимые вещи»: создание списка любимых развлечений и любимых лакомств.

#### **Тема 1.8. Рисование с помощью модуля Turtle.**

Теория. Понятие «модуль». Модуль turtle. Импортирование модуля. Создание холста. Перемещение черепашки.

Практика. Проект «Я рисую»: рисование изображения, состоящего из линий с помощью команд модуля на холсте.

### **Раздел 2. Основы языка Python**

#### **Тема 2.1. Условный оператор**

Теория. Понятие «условный оператор», «вложенные команды», «оператор сравнения». Конструкция if и её синтаксис. Операторы сравнения: <, >, >=, <=, !=, ==. Структура программы. Конструкция if-else. Команды if и elif.

Практика. Решение задач на обработку условного оператора и операторов сравнения.

#### **Тема 2.2. Объединение условий**

Теория. Логические операторы: and, or, not. Порядок выполнения операций. Переменные без значения – None.

Практика. Проект «Калькулятор»: создание приложения по определенным условиям.

### **Тема 2.3. Использование цикла for**

Теория. Понятие «цикл», «цикл со счётчиком». Конструкция for и её синтаксис.

Практика. Проект «Таблица умножения»: создание приложения по определенным условиям.

### **Тема 2.4. Использование цикла while**

Теория. Понятие «цикл с предусловием». Конструкция while и её синтаксис. Заикливание и выход из цикла с помощью команды break.

Практика. Проект «Бомба взорвалась!». Написание программы по определенным условиям.

### **Тема 2.5. Решение задач на циклические алгоритмы**

Теория. Виды циклов и их конструкции.

Практика. Решение задач на применение циклов for и while.

### **Тема 2.6. Применение функций**

Теория. Понятие «функция», «параметр функции», «значение функции». Строение функции: имя, аргумент, тело. Создание и вызов функции. Переменные и область видимости.

Практика. Решение задач на отработку понятия «функция», её строение и синтаксис.

### **Тема 2.7. Использование встроенных функций**

Теория. Функции: abs, bool, dir, eval, exes, float, int, len, max, min, range, sum.

Практика. Решение задач на отработку функций.

### **Тема 2.8. Графика с модулем tkinter**

Теория. Модуль tkinter и его возможности.

Практика. Проект «Анимация»: отрисовывание объекта на холсте и программирование анимации.

## **Раздел 3. Итоговый проект.**

Практика. Разработка собственного проекта на тему, согласованную с учителем.

## **7. Условия реализации программы**

Комплекс условий реализации программы:

Аппаратное и техническое обеспечение:

а) Рабочее место учащегося

- компьютер или ноутбук с выходом в сеть Интернет;

- установленный на каждый ПК интерпретатор языка программирования Python 3 с установленным IDE PyCharm.

б) Рабочее место наставника

- компьютер или ноутбук с выходом в сеть Интернет;

- установленный на ПК интерпретатор языка программирования Python 3 с установленным IDE PyCharm;

- технические средства обучения (ТСО) (мультимедийное устройство).

Методы обучения:

- объяснительно-иллюстративный (беседы, объяснения, дискуссии);

- репродуктивный (деятельность обучаемых носит алгоритмический характер, выполняется по инструкциям, предписаниям, правилам в аналогичных, сходных с показанным образцом ситуациях);

- метод проблемного изложения;

- эвристический (метод обучения заключается в организации активного поиска решения выдвинутых в обучении (или самостоятельно сформулированных) познавательных задач в ходе подготовки и реализации творческих проектов);

- исследовательский.

## **Кадровые условия реализации программы**

Программу реализует педагог дополнительного образования, который обладает навыками программирования на языке Python, владеет проектным мышлением и умеет организовать групповую проектную деятельность учащихся и руководить ею.

## **8. Перечень рекомендуемых источников**

1. Бриггс, Джейсон, Python для детей. Самоучитель по программированию / Джейсон Бриггс ; пер. с англ. Станислава Ломакина ; [науч. ред. Д. Абрамова]. — М. : Манн, Иванов и Фербер, 2017. — 320 с.
2. Великович Л., Цветкова М. Программирование для начинающих. — М.: Бином, 2008.
3. Программирование для детей. Перевод с английского Станислава Ломакина, Москва, «Манн, Иванов и Фербер», 2015 г.
4. Сайт [pythonworld.ru](http://pythonworld.ru) — «Python 3 для начинающих».
5. Сайт [pythontutor.ru](http://pythontutor.ru) — «Питонтьютор».