

ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ИТ-КУБ ГОРОДА КОСТРОМЫ



Дополнительное образование

УЧЕБНАЯ ПРОГРАММА ПО НАПРАВЛЕНИЮ

Программирование на Python

8-9 классы

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 8 - 9 классов.

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	
1	Введение в язык программирования Python	26	11	15	
1.1	Устройство языка Python. Среда разработки.	2	1	1	Наблюдение
1.2	PEP8. Ввод-вывод. Переменные	2	1	1	Наблюдение
1.3	Типы данных. Операции над числами	2	1	1	Опрос
1.4	Строки	4	2	2	Опрос
1.5	Списки	4	2	2	Опрос
1.6	Numpy	3	1	2	Опрос
1.7	Списочные выражения. Методы split() и join	3	1	2	Опрос
1.8	Кортежи и словари	3	1	2	Опрос
1.9	Множества	3	1	2	Опрос
2	Основы языка Python	42	11	31	
2.1	Условный оператор	3	1	2	Опрос
2.2	Сложные условия. Вложенные структуры. Логические операции и их свойства	3	1	2	Самостоятельная работа
2.3	Использование цикла for	4	1	3	Опрос
2.4	Использование цикла while	4	1	3	Самостоятельная работа
2.5	Использование встроенных функций	3	1	2	Опрос
2.6	Применение функций	4	1	2	Самостоятельная работа
2.7	Встроенные модули	3	1	2	Опрос
2.8	Файловая система. Файлы в Python	3	1	2	Самостоятельная работа
2.9	Обработка коллекций. Поточковый ввод sys.stdin	4	1	3	Опрос
2.10	Решение задач из ОГЭ	5	1	4	Опрос
3	Итоговый проект	6	1	5	Защита проекта
	Итого:	68	22	46	

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. PEP8. Ввод-вывод. Переменные

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

Практика. Решение задач на применение функций print() и input().

Тема 1.3. Типы данных. Операции над числами

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

Практика. Проект «Сумматор».

Тема 1.4. Строки

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

Практика. Решение задач на обработку операций со строками.

Тема 1.5. Списки

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

Практика. Решение задач на обработку операций со списками.

Тема 1.6. Массивы в NumPy.

Теория. Модуль NumPy. Размерность массива, индексация в массивах, массовые операции.

Практика. Решение задач по теме.

Тема 1.7. Списочные выражения. Методы split() и join()

Теория. Понятие «списочное выражение». Порядок создания выражений.

Практика. Решение задач на использование списочных выражений в аргументах методов split и join. Считывание значений, введенных одной строкой

Тема 1.8. Кортежи и словари

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

Практика. Проект «Азбука Морзе»

Тема 1.9. Множества.

Теория. Понятие «множество». Операции над множествами, сравнение двух множеств.

Практика. Решение задач на обработку операций с множествами.

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. Логические операторы: and, or, not. Порядок выполнения операций.

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

Тема 2.2. Сложные условия. Вложенные структуры. Логические операции и их свойства

Теория. Вложенные условия, операции над строками, команда in.

Практика. Решение задач по теме. Проект «Квест»

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. Решение задач по теме.

Тема 2.5. Использование встроенных функций

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

Практика. Решение задач на отработку функций.

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. Применение модулей

Теория. Понятие «модуль». Импорт модуля в программу. Полезные модули: `random`, `time`, `pickle`.

Практика. Решение задач на применение модулей.

Тема 2.8. Файловая система. Файлы в Python

Теория. Общие сведения о файлах, перевод строки. Файловый путь. Относительные и абсолютные пути. Кодировки файлов. Типичные операции с файлами (чтение, запись).

Практика. Решение задач по теме.

Тема 2.9. Обработка коллекций.. Поточковый ввод `sys.stdin`

Теория. Интегрируемые объекты. Операторы `filter` и `map`. Функции `max/min/sort` и использование ключа сортировки. Проверка коллекций; `all`, `any`.

Практика. Решение задач по теме.

Тема 2.10. Решение задач из ОГЭ

Теория. Типы задач из ОГЭ, решаемых с помощью программирования. Разбор.

Практика. Решение задач по теме.

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 — «Python 3 для начинающих».
5. Сайт pythontutor.ru — «Питонтьютор».