

Муниципальное общеобразовательное учреждение  
«Гимназия им. В.А. Надькина»

Рассмотрено на заседании  
научно-методического совета  
№ 5 от 06.12.2020 г.

Утверждено приказом  
директора МОУ  
«Гимназия им. В.А. Надькина»  
№116/1-26-421 от 14.12.2020 г.

**Дополнительная общеобразовательная общеразвивающая программа  
технической направленности  
«Основы программирования»**

Возраст учащихся: 11-13 лет

Срок реализации: 3 года

Составитель: Голубцов Андрей  
Юрьевич, педагог дополнительного  
образования

2020 год

## I. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Рабочая программа предназначена для изучения языка программирования Python на начальном уровне в 5-7 классах. Языком для изучения выбран Python. Выбор обусловлен достоинствами языка:

- кроссплатформенность и бесплатность;
- простой синтаксис и богатые возможности;
- расширенная стандартная библиотека, возможность, в будущем, разработки промышленных приложений.

Язык Python поддерживается на всех олимпиадах для школьников всероссийского уровня. С 2015 года во всех вариантах КИМ ЕГЭ по информатике примеры программ записываются, в том числе и на языке Python.

Цель: формирование базовых понятий структурного программирования, развитие логики обучающихся. Курс программы рассчитан на 34 часа с периодичностью занятий 1 раз в неделю для 5-6 классов и 68 часов с периодичностью 2 раза в неделю для 7 класса; возрастная группа: 11-13 лет.

Форма обучения индивидуально-групповая, включающая в себя следующие виды деятельности: беседы, лекции, практические занятия.

Задачи курса:

1. формирование знаний об алгоритмических конструкциях, логических значениях и операциях;
2. знакомство с языком программирования Python и основными алгоритмическими структурами — линейной, условной и циклической;
3. владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;

Планируемые результаты освоения программы:

Личностные	• освоение социальной роли обучающегося, развитие мотивов учебной деятельности и формирование личностного смысла
------------	------------------------------------------------------------------------------------------------------------------

	<p>учения;</p> <ul style="list-style-type: none"> <li>• развитие самостоятельности и личной ответственности за свои поступки, в том числе в процессе учения;</li> <li>• формирование целостного, социально ориентированного взгляда на мир в его органичном единстве и разнообразии природы, народов, культур и религий;</li> <li>• овладение начальными навыками адаптации в динамично изменяющемся и развивающемся мире;</li> <li>• сформированность мировоззрения, соответствующего современному уровню развития науки и техники;</li> <li>• осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов.</li> </ul>
Метапредметные:	<ul style="list-style-type: none"> <li>• умение самостоятельно контролировать своё время;</li> <li>• осуществлять самостоятельный контроль, адекватно оценивать правильность выполнения действия и вносить необходимые коррективы как в конце действия, так и по ходу его реализации.</li> </ul>
регулятивные	
познавательные	<ul style="list-style-type: none"> <li>• выявлять сходства и различия в социокультурных, географических особенностях разных стран;</li> <li>• извлекать информацию, представленную в разных формах (текст, иллюстрация, схема, таблица);</li> <li>• анализировать объекты с целью выделения признаков;</li> <li>• устанавливать причинно-следственные связи;</li> <li>• осуществлять анализ и синтез полученной информации;</li> <li>• формулировать проблему и самостоятельно находить способы её решения;</li> <li>• работать с источниками, использовать алгоритм критического анализа источника.</li> </ul>
коммуникативные	<ul style="list-style-type: none"> <li>• учитывать разные мнения и стремиться к координации различных позиций в сотрудничестве;</li> <li>• формулировать и аргументировать собственное мнение и позицию;</li> <li>• высказывать и обосновывать свою точку зрения;</li> <li>• воспринимать иные точки зрения, быть готовым корректировать свою;</li> <li>• выступать перед аудиторией сверстников с сообщениями,</li> </ul>

	докладами, рефератами.
Предметные	<ul style="list-style-type: none"> <li>• владение навыками алгоритмического мышления и понимание необходимости формального описания алгоритмов;</li> <li>• владение стандартными приёмами написания программы для решения стандартной задачи с использованием основных конструкций программирования и отладки таких программ на языке Python;</li> <li>• владение навыками и опытом разработки программ на Python, включая тестирование и отладку программ;</li> <li>• владение элементарными навыками формализации прикладной задачи.</li> </ul>

*Способ определения результативности:* опрос, диагностические задания, защита проектов, активность на занятиях.

*Для отслеживания результативности* можно использовать: педагогический мониторинг, включающий в себя тесты, контрольные.

*Мониторинг образовательной деятельности:* оценка практических работ, анализ опроса, анализ самостоятельных работ.

*Виды контроля:* текущий, итоговый.

*Форма итога:* защита проекта.

## II. УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН<sup>1</sup>

№	Тема	Кол-во часов	Кор-ка часов
<b>5 класс</b>			
<b>Раздел 1. Начало работы</b>			
1	Техника безопасности. Вводное занятие	1	
2	Знакомство с Python	1	
3	Знакомство с IDLE	1	

<sup>1</sup> Педагог дополнительного образования вправе увеличить количество часов на освоение отдельных разделов и произвести перестановку тем на основании результатов внутреннего мониторинга для освоения отдельных тем, для закрепления навыков и т.п., при обязательном обеспечении достижения планируемых результатов и элементов содержания программы по предмету.

<b>Раздел 2. Типы, переменные и основы ввода-вывода</b>			
1	Строки и кавычки	1	
2	Escape-последовательности в строках	1	
3	Сцепление и повторение строк	1	
4	Работа с числами	1	
5	Переменные	1	
6	Получение пользовательского ввода	1	
7	Применение строковых методов	1	
8	Правильно выбранный тип	1	
9	Конвертация значений	1	
<b>Раздел 3. Ветвление, циклы с оператором while и псевдокод</b>			
1	Генерирование случайных чисел	2	
2	Условные конструкции с if	2	
3	Конструкции if с условием else	2	
4	Использование выражения elif	2	
5	Создание циклов с использованием while	2	
6	Бесконечные циклы	1	
7	Значения как условия	1	
8	Намеренное создание бесконечных циклов	1	
9	Составные условия	1	
10	Планирование программ	1	
<b>Раздел 4. Циклы с оператором for, строки и кортежи</b>			
1	Применение циклов for	2	
2	Счёт с помощью цикла for	2	
3	Операторы и функции для работы с последовательностями	2	
4	Индексация строк	1	
<b>6 класс</b>			
<b>Раздел 1. Циклы с оператором for, строки и кортежи</b>			

1	Техника безопасности. Вводное занятие	1	
2	Неизменяемость строк	1	
3	Создание новой строки	2	
4	Срезы строк	2	
5	Использование кортежей	2	
<b>Раздел 2. Списки и словари</b>			
1	Использование списков	3	
2	Применение списочных методов	3	
3	Когда использовать кортежи, а когда – списки	1	
4	Вложенные последовательности	3	
5	Распределённые ссылки	1	
6	Использование словарей	3	
<b>Раздел 3. Функции</b>			
1	Создание функций	3	
2	Параметры и возвращаемые значения	3	
3	Именованные аргументы и значения параметров по умолчанию	3	
4	Использование глобальных переменных и констант	3	
<b>7 класс</b>			
<b>Раздел 1. Файлы и исключения</b>			
1	Техника безопасности. Вводное занятие	1	
2	Чтение текстового файла	2	
3	Запись в текстовый файл	3	
4	Хранение структурированных данных в файлах	3	
5	Обработка исключений	3	
<b>Раздел 2. Программные объекты</b>			
	Основы объектно-ориентированного подхода	1	
1	Создание классов, методов и объектов	3	

2	Применение конструкторов	3	
3	Применение атрибутов	3	
4	Применение атрибутов класса и статических методов	3	
5	Инкапсуляция объектов	1	
6	Применение закрытых атрибутов и методов	3	
7	Управление доступом к атрибутам	3	
<b>Раздел 3. Объектно-ориентированное программирование</b>			
1	Отправка и прием сообщений	3	
2	Сочетание объектов	4	
3	Создание новых классов с помощью наследования	1	
4	Расширение класса через наследование	4	
5	Переопределение унаследованных методов	4	
6	Полиморфизм	1	
7	Создание модулей	4	
<b>Раздел 4. Программирование беспилотных летательных аппаратов</b>			
1	Проведение полётов в ручном режиме	1	
2	Программирование взлёта и посадки беспилотного летательного аппарата	1	
3	Выполнение команд «разворот», «изменение высоты», «изменение позиции»	3	
4	Выполнение группового полёта вручную	1	
5	Выполнение позиционирования по меткам	3	
6	Программирование группового полёта	3	
7	Программирование роевого взаимодействия	3	

### III. СОДЕРЖАНИЕ ПРОГРАММЫ

№	Название раздела	Количество часов
---	------------------	------------------

		<b>Всего</b>	<b>Теория</b>	<b>Практика</b>
<b>5 класс</b>				
1	Начало работы	3	3	0
2	Типы, переменные и основы ввода-вывода	9	3	6
3	Ветвление, циклы с оператором while и псевдокод	15	5	10
4	Циклы с оператором for, строки и кортежи	7	2	5
<b>ИТОГО</b>		34	13	21
<b>6 класс</b>				
1	Циклы с оператором for, строки и кортежи	8	3	5
2	Списки и словари	14	4	10
3	Функции	12	4	8
<b>ИТОГО</b>		34	11	23
<b>7 класс</b>				
1	Файлы и исключения	12	3	9
2	Программные объекты	20	6	14
3	Объектно-ориентированное программирование	21	7	14
4	Программирование беспилотных летательных аппаратов	15	6	9
<b>ИТОГО</b>		68	22	46

### 5 класс

**Раздел 1. Знакомство с языком Python:** общие сведения о языке; среда разработки IDLE; программирование в интерактивном режиме;



программирование в сценарном режиме; использование комментариев.

**Раздел 2. Типы, переменные и основы ввода-вывода:** кавычки внутри строк; создание строк в тройных кавычках; вставка табуляционного отступа; звук системного динамика; сцепление строк; числовые типы данных; применение математических операторов; создание переменных; применение функции input; создание новых строк с помощью строковых методов; составные операторы присвоения; получение пользовательского ввода.

**Раздел 3. Ветвление, циклы с оператором while и псевдокод:** применение функций randint и randrange; создание условных конструкций; операторы сравнения; условия else и elif; цикл while; трассировка программ; истинные и ложные значения; команды break и continue; логические операторы; алгоритмы на псевдокоде.

**Раздел 4. Циклы с оператором for, строки и кортежи:** создание цикла for; счет с помощью цикла for; применение функции len к строкам; применение оператора in к строкам; индексация строк.

## 6 класс

**Раздел 1. Циклы с оператором for, строки и кортежи:** создание констант; создание срезов строк; создание кортежей; применение функции len к кортежам; применение оператора in к кортежам; индексация кортежей; срезы кортежей; сцепление кортежей; получение пользовательского ввода.

**Раздел 2. Списки и словари:** создание списка; применение функции len к спискам; применение оператора in к спискам; индексация списков; срезы списков; сцепление списков; удаление элемента списка; удаление среза списка; применение списочных методов; создание вложенных последовательностей; создание словарей.

**Раздел 3. Функции:** объявление функции; документирование функции; передача данных с помощью параметров; инкапсуляция; функции, которые и принимают и возвращают значения; позиционные параметры и позиционные аргументы; позиционные параметры и именованные аргументы; чтение глобальной переменной внутри функции.

## 7 класс

**Раздел 1. Файлы и исключения:** чтение текстового файла; посимвольное чтение строк; чтение всех строк файла в список; перебор строк файла; запись строк в файл; запись списка строк в файл; консервация данных и запись в файл; чтение и расконсервация данных из файла; применение конструкций try/except; типы исключений; добавление блока else.

**Раздел 2. Программные объекты:** объявление класса; объявление метода; создание объекта; вызов метода; создание конструктора; инициализация и доступ к атрибутам; вывод объекта на экран; создание и доступ к атрибутам класса; создание и вызов статического метода; создание и доступ к закрытым атрибутам; создание и доступ к закрытым методам; создание и доступ к свойствам.

**Раздел 3. Объектно-ориентированное программирование:** отправка и прием сообщений; создание базового класса; наследование от базового класса; расширение производного класса; применение производного класса; переопределение методов базового класса; вызов методов базового класса; применение импортированных функций и классов.

**Раздел 4. Программирование беспилотных летательных аппаратов:** полёты на квадрокоптерах в ручном режиме; основы программирования квадрокоптеров на языке Python; тестирование написанного кода в режимах взлёта и посадки; тестирование программного кода в режимах разворота, изменения высоты и позиции; выполнение группового полёта на квадрокоптере в ручном режиме; основы позиционирования квадрокоптеров; основы программирования роя квадрокоптеров; выполнение группового полета в

автоматическом режиме.

#### **IV. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ**

Приемы и методы организации занятий:

- а) словесные методы (рассказ, беседа, инструктаж, чтение справочной литературы);
- б) наглядные методы (демонстрации мультимедийных презентаций);
- в) практические методы (упражнения, задачи).

Логический аспект:

- а) индуктивные методы, дедуктивные методы;
- б) конкретные и абстрактные методы, синтез и анализ, сравнение, обобщение, абстрагирование, т.е. методы как мыслительные операции.

Методы стимулирования мотива интереса к занятиям: познавательные задачи, учебные дискуссии, опора на неожиданность, создание ситуации новизны, ситуации гарантированного успеха и т.д.

**Материально-техническое обеспечение:** компьютерный класс, оснащенный персональными компьютерами; канал Интернет; среда разработки IDLE; квадрокоптеры «Пионер Мини».

## СПИСОК ЛИТЕРАТУРЫ

1. Майк МакГрат Программирование на Python для начинающих: Методические материалы / Майк МакГрат. – М.: Эксмо, 2015. – 212 с.
2. Федоров Д. Ю. Основы программирования на примере языка Python: Учебное пособие / Д.Ю. Федоров. –СПбГЭУ, 2018. – 167 с.
3. Сэнд У., Сенд К. Hello World! Занимательное программирование на языке Python: Методические рекомендации/ У. Сэнд, К. Сенд. - М.: – 2016. – 185 с.

## Тест итогового контроля «Язык Python»

**Задание:** Добавьте каждое предложение по смыслу. За каждый правильный ответ вы набираете 1 балл.

1. Программа Python называется ...
2. Расширение файла Python – as. ...
3. Переменная в Python – это ...
4. Регистр букв в идентификаторах значение ...
5. Выражение в Python – это ...
6. Символ # в Python обозначает ...
7. ... в Python это тип данных для вещественных чисел, встроенный в Python по-умолчанию.
8. Операция  $3**4$  - это
9. 345 - ... тип данных.
10. Операция  $46\%10$  – это ...
11. Функция `round(d)` – это ...
12. Функция `input()` – предназначена для ...
13. Для вывода данных есть функция в Python - ...
14. ... в Python это логический тип данных, встроенный в Python по-умолчанию.
15. Строки – это ...
16. `A='pri', s='vet'`. `A+s` – это ...
17. `E='no'`. `E*5` – это ...
18. К элементу в строке можно обратиться по ...
19. `s='asdfgh'`

`print(s[-1])`. Программа выведет ...

20. `s='asdfgh'`

`print(s[2:4])`. Программа выведет ...

21. Функция `len(строка)` – возвращает ...
22. Списки – это ...
23. Пример списка - ...
24. Словари – это ...
25. Пример словаря - ...
26. Условный оператор в Python - ...
27. Цикл `for` называется циклом ...
28. Переведите конструкцию языка

```
S=[1,2,3]
```

```
for I in s:
```

```
print(I*4)
```

29. Функция `range()` переводится как ...

30. Переведите конструкцию языка

```
S=0
```

```
While S<10:
```

```
print(S)
```

```
S=S+1
```

31. Функции — это ...

32. Локальные переменные объявлены ...