

## Пояснительная записка

На сегодняшний день в различные отрасли производства требуются квалифицированные сотрудники в сфере ИТ-технологий, одним из направлений в которой является программирование различных информационных систем и технических устройств. В рамках изучения данной программы, обучающиеся приобретут навыки составления алгоритмов решения широкого класса задач с помощью языка программирования Python, что является необходимым условием подготовки современных учащихся.

Особая роль отводится широко представленной в курсе системе рефлексивных заданий, что помогает осознанию учащимися связки «разработка программного продукта – освоение инструментария программирования».

Содержание обучения, представленное в программе «Увлекательное программирование», позволяет вести обучение в режиме актуального познания. Практическая направленность курса на создание внешних образовательных продуктов – блок-схем, алгоритмов, программ – способствует выявлению фактов, которые невозможно объяснить на основе имеющихся у учащихся знаний.

**Направленность:** техническая.

**Возраст детей:** 13-15 лет.

**Актуальность программы** состоит в том, что активизация познавательного процесса позволяет учащимся более полно выражать свой творческий потенциал и реализовывать собственные идеи в изучаемой области знаний, создаёт предпосылки по применению информационных компетенций в других учебных курсах, а также способствует возникновению дальнейшей мотивации, направленной на освоение профессий, связанных с разработкой программного обеспечения.

**Ключевой особенностью курса** является его направленность на формирование у учащихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования.

**Цель:** Формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

### Задачи

#### 1. Образовательные:

- научить составлять и читать блок-схемы;
- освоить основы программирования на языке Python (основные конструкции языка программирования, типы данных, функции, отладка, тестирование программ и т. д.);
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python.

#### 2. Воспитательные:

- сформировать культуру поведения во время занятий и совместной продуктивной деятельности;
- повысить уровень самостоятельности в обучении;
- сформировать культуру занятий, направленную на воспитание личностных и социальных качеств.

#### 3. Развивающие:

- развивать познавательные процессы (внимание, восприятие, логическое мышление, память),
  - способствовать развитию критического, системного, алгоритмического, творческого мышления и способности к самореализации;
  - развивать умение работать с компьютерными программами и дополнительными источниками информации.

### Условия реализации программы.

**Материально-техническая база:** наличие компьютерного класса (ноутбуки, проектор, интерактивная доска, ЛВС и выход в сеть интернет), учебные пособия, дидактический материал.

**Формы организации детей:** групповые, индивидуальные.

Форма обучения - очная, очно-заочная с применением электронного обучения и дистанционных образовательных технологий.

Уровень освоения программы – **базовый**.

Программа рассчитана **на 1 год обучения, 68 часов**.

Занятия проводятся **1 раз в неделю по 2 часу** для 7-8-х классов.

Наполняемость группы – **10 человек**.

### **Планируемые результаты освоения программы.**

#### **Личностные**

По окончании курса у учащийся сформируется:

- способность к саморазвитию и самообразованию средствами информационных технологий на основе, приобретённой благодаря иллюстративной среде программирования мотивации к обучению и познанию;
- коммуникативная компетентность в общении и сотрудничестве со сверстниками в процессе образовательной;
- целостность мировоззрения, соответствующего современному уровню развития информационных технологий;
- умение самостоятельно реализовывать проекты, связанные с программированием.

#### **Метапредметные**

Разовьют умение:

- ставить учебные цели;
- формулировать достигнутый результат;
- планировать свою самостоятельную учебно-познавательную деятельность; выбирать индивидуальную траекторию достижения учебной цели;
- определять подходы и методы для достижения поставленной цели;
- отбирать необходимые средства для достижения поставленной цели;
- осуществлять самооценку промежуточных и итоговых результатов своей самостоятельной учебно-познавательной деятельности;
- проводить рефлексию своей учебно-познавательной деятельности.

#### **Предметные**

В рамках курса «Увлекательное программирование» учащиеся овладевают следующими знаниями, умениями и способами деятельности:

- умеют составлять алгоритмы для решения задач;
- умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- владеют основными навыками программирования на языке Python;
- умеют отлаживать и тестировать программы, написанные на языке Python.

#### ***Способы оценивания уровня достижений учащихся***

Предметом диагностики и контроля в курсе «Увлекательное программирование» являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Качество внешней образовательной продукции желательно оценивать по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
- программа должна выполнять поставленные задачи;
- по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).
- Проверка достигаемых учащимися результатов производится в следующих формах:
- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
- текущая диагностика и оценка педагогом деятельности учащихся;

- итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальными и возрастными особенностями;
- Итоговый контроль проводится в конце всего курса. Он состоит из тестирования и решения практической задачи, защиты творческого проекта.

## Содержание программы

Один год обучения – 68 часов.

### **Вводное занятие (1 час).**

Теория: Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Опрос. Тренинг на командообразование.

### **Раздел 1. Знакомство с языком Python (3 часа)**

Теория: Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с Python Тест № 1. Знакомство с языком Python

### **Раздел 2. Переменные и выражения (5 часов)**

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой Практическая работа 2.2. Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами Тест № 2. Выражения и операции.

### **Раздел 3. Условные предложения (10 часов)**

Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление Практическая работа 3.4. "Условные операторы"

Самостоятельная работа № 1. Решение задач по теме "Условные операторы". Зачетная работа № 1. "Составление программ с ветвлением".

Тест № 3. "Условные операторы".

### **Раздел 4. Циклы (16 часов)**

Теория: Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. "Числа Фибоначчи" Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом. Самостоятельная работа № 2 "Составление программ с циклом"

Тест № 4. Циклы

Творческая работа № 1. "Циклы"

### **Раздел 5. Функции (10 часов)**

Теория: Создание функций. Параметры и аргументы. Локальные и глобальные переменные. Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи. Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные  
 Практическая работа 5.3. Решение задач с использованием функций Практическая работа  
 5.4. Рекурсивные функции

Самостоятельная работа № 3 по теме "Функции" Тест № 5. Функции

### **Раздел 6. Строки - последовательности символов (8 часов)**

Теория: Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

### **Раздел 7. Сложные типы данных (10 часов)**

Теория: Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range. Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python. Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками. Тест № 7. Списки

### **Раздел 8. Стиль программирования и отладка программ (5 часов)**

Стиль программирования. Отладка программ.

Зачет по курсу «Программирование на языке Python»

## **Учебно-тематический план**

№ п/п	Наименование тем	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
	<b>Инструктаж по технике безопасности в компьютерном классе</b>	<b>1</b>	<b>1</b>		Опрос
<b>1.</b>	<b>Раздел 1. Знакомство с языком Python</b>	<b>3</b>	<b>1</b>	<b>2</b>	Тестирование, решение практических задач
1.1	Занятие 1. Общие сведения о языке Практическая работа: Установка программы Python	1		1	
1.2.	Занятие 2. Режимы работы Практическая работа: Режимы работы с Python	1		1	
1.3	Тест № 1. Знакомство с языком Python	1	1		
<b>2.</b>	<b>Раздел 2. Переменные и выражения</b>	<b>5</b>	<b>1</b>	<b>5</b>	Тестирование, решение практических задач
2.1	Занятие 3. Переменные Практическая работа: Работа со справочной системой.	1		1	

	Практическая работа: Переменные.				
2.2	Занятие 4. Выражения Практическая работа: Выражения	1		1	
2.3	Занятие 5. Ввод и вывод	1		1	
2.4	Занятие 6. Задачи на элементарные действия с числами Практическая работа 2.5. Задачи на элементарные действия с числами	1		1	
2.5	Тест № 2. Выражения и операции.	1	1		
<b>3</b>	<b>Раздел 3. Условные предложения</b>	<b>10</b>	<b>3</b>	<b>7</b>	
3.1	Занятие 7. Логические выражения и операторы. Практическая работа: Логические выражения	1		1	
3.2	Занятие 8. Условный оператор Практическая работа: «Условный оператор»	1		1	
3.3	Занятие 9. Множественное ветвление Практическая работа: Множественное ветвление	1		1	
3.4	Занятие 10. Реализация ветвления в языке Python. Практическая работа: «Условные операторы»	2	1	1	
3.5	Самостоятельная работа № 1 по теме «Условные операторы».	2	1	1	
3.6	Занятие 11. Зачетная работа № 1. "Составление программ с ветвлением".	2		2	
3.7	Тест № 3. "Условные операторы".	1	1		
<b>4</b>	<b>Раздел 4. Циклы</b>	<b>16</b>	<b>6</b>	<b>10</b>	
4.1.	Занятие 12. Оператор цикла с условием Практическая работа "Числа Фибоначчи"	2	1	1	
4.2.	Занятие 13. Оператор цикла for Практическая работа Решение задачи с циклом for.	2	1	1	
4.3.	Занятие 14. Вложенные циклы Практическая работа: Реализация циклических алгоритмов	2	1	1	
4.4.	Занятие 15. Случайные числа Практическая работа: Случайные числа	1		1	
4.5.	Занятие 16. Примеры решения задач с циклом	3	1	2	
					Тестирование, Решение практических задач
					Тестирование, решение практических задач, творческая работа

	Практическая работа: Решение задач с циклом.				
4.6	Самостоятельная работа № 2 "Составление программ с циклом"	2		2	
4.7	Тест № 4. Циклы	2	1	1	
4.8	Занятие 17. Творческая работа № 1. "Циклы"	2	1	1	
<b>5</b>	<b>Раздел 5. Функции</b>	<b>10</b>	<b>4</b>	<b>6</b>	
5.1.	Занятие 18. Создание функций Практическая работа Создание функций	2	1	1	Тестирование, решение практических задач
5.2.	Занятие 19. Локальные переменные Практическая работа Локальные переменные	1		1	
5.3.	Занятие 20. Примеры решения задач с использованием функций Практическая работа Решение задач с использованием функций	2	1	1	
5.4	Самостоятельная работа № 3 по теме "Функции"	1		1	
5.5	Занятие 21. Рекурсивные функции Практическая работа 5.4. Рекурсивные функции	2	1	1	
5.6	Тест № 5. Функции	2	1	1	
<b>6</b>	<b>Раздел 6. Строки - последовательности символов</b>	<b>8</b>	<b>3</b>	<b>5</b>	
6.1.	Занятие 22. Строки Практическая работа: Строки	2	1	1	Решение практических задач
6.2.	Занятие 23. Срезы строк	2	1	1	
6.3	Занятие 24. Примеры решения задач со строками Практическая работа: Решение задач со строками.	4	1	3	
<b>7</b>	<b>Раздел 7. Сложные типы данных</b>	<b>10</b>			
7.1.	Занятие 25. Списки Практическая работа 7.1 Списки. Занятие 26. Срезы списков	2	1	1	Тестирование, Решение практических задач
7.2	Занятие 27. Списки: примеры решения задач Практическая работа 7.2. Решение задач со списками	3	1	2	
7.3	Занятие 28. Матрицы	2	1	1	
7.4	Тест № 7. Списки	1	1		
7.5	Занятие 29. Кортежи. Занятие 30. Введение в словари	1	1		
7.6	Занятие 31. Множества в языке Python	1		1	
7.7					

<b>8</b>	<b>Раздел 8. Стил программирования и отладка программ</b>	<b>5</b>	<b>2</b>	<b>3</b>	Решение практических задач, презентация проекта, рефлексия.
8.1	Занятие 32. Стил программирования. Занятие 33. Отладка программ	2	1	1	
8.2					
8.3	Занятие 34. Зачет по курсу «Увлекательное программирование»	3	1	2	
8.3	Занятие 35. Что дальше?	2	2		
	<b>ВСЕГО</b>	<b>156</b>	<b>45</b>	<b>111</b>	



### Список литературы

1. Васильев, А. Н. Python на примерах: практический курс /А. Н. Васильев - Наука и Техника, 2019 - 432 с.
2. Прохоренок, Н. А. Python 3: самое необходимое: практический курс / Н. А. Прохоренок, В. А. Дронов - БХВ-Петербург, 2019 - 608 с.
3. Гэддис, Т. Начинаем программировать на Python: учебник/Т. Гэддис - БХВ-Петербург, 2019 - 768 с.
4. Седжвик, Р. Программирование на языке Python /Р. Седжвик, К. Уэйн, Р. Дондеро - Вильямс, 2017 - 736 с.
5. Лутц, Марк Python. Справочник. – М.: Вильямс, 2019.
6. Официальный сайт программы [Электронный ресурс] – Режим доступа: <https://docs.python.org/>, свободный.
7. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.
8. Сайт / справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.
9. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.
10. Сайт / Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431> , свободный.
11. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu> , свободный.