

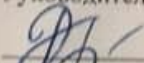


МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА № 14
ПРЕДГОРНОГО МУНИЦИПАЛЬНОГО ОКРУГА
СТАВРОПОЛЬСКОГО КРАЯ

357355 РФ Ставропольский край Предгорный район пос. Пятигорский ул. Новая, 1 А,
тел: 8 (87961) 48-2-34, 48-7-47 / факс: 8 (87961) 48-2-34, 48-7-47

СОГЛАСОВАНО

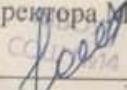
Руководитель Центра ТР

 Р.Х. Резанова

от «29» августа 2022 г.

УТВЕРЖДАЮ

и.о. директора МБОУ СОШ № 14

 Ю.Ю. Карацева

пр. №190 от «29» августа 2022 г.



Дополнительная
общеразвивающая программа
Центра образования естественно-научной и
технологической направленностей «Точка Роста»
«Основы программирования»
возраст учащихся: 15 – 17 лет
срок реализации: 1 год

Составитель:
Вернохаев Г.Г.
учитель математики и информатики
МБОУ СОШ № 14

2022 г.

Пояснительная записка

Одна из задач профильной школы – содействовать воспитанию нового поколения, отвечающего по своему уровню развития и образу жизни условиям информационного общества. Для этого учащимся необходимо анализировать информацию, выявлять в ней факты и проблемы, самостоятельно ставить задачи, структурировать и преобразовывать информацию, использовать ее для решения учебных и жизненных задач.

Программа кружка «Программирование на языке Python» предназначена для учащихся 15-17 лет, рассчитан на 76,5 часов, которые проводятся в течение учебного года по 1 и 1,25 часов в неделю. Концентрированное изучение курса позволяет учащимся более полно выявить свои способности в изучаемой области знаний, создать предпосылки по применению освоенных умений в других учебных курсах, подготовить себя к осознанному выбору профессий, предусматривающих программирование.

Курс включает в себя практическое освоение языка программирования, знакомство учащихся с ролью программного обеспечения и его видами; нацелен на формирование целостного представления об организации данных для эффективной алгоритмической обработки; на развитие логического мышления и реализацию математических способностей учащихся в ходе составления программ на языке программирования.

Основа курса – личностная, практическая и продуктивная направленность занятий. Одна из целей обучения информатике – предоставить ученикам возможность личностного самоопределения и самореализации по отношению к стремительно развивающимся информационным технологиям и ресурсам.

Цели:

- Познакомить учащихся с ролью программного обеспечения и его видами.
- Сформировать целостное представление об организации данных для эффективной алгоритмической обработки.
- Развитие логического мышления.
- Реализация математических способностей учащихся в ходе составления программ на языке программирования.

Задачи:

Обучающие:

- Познакомить учащихся с основными алгоритмическими конструкциями и правилами их записи, с основными способами организации данных.

- Научить учащихся составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций.
- Научить распознавать необходимость применения той или иной алгоритмической конструкции при решении задач.
- Научить организовывать данные для эффективной алгоритмической обработки.
- Научить учащихся разрабатывать алгоритмы и реализовывать их на языке программирования Python .
- Научить учащихся осуществлять отладку и тестирование программы.

Развивающие:

- формировать новый тип мышления – операционный, который направлен на выбор оптимальных решений;
- предоставление возможности узнать новое в области компьютерного программирования;
- формирование представления о роли компьютерного программирования в развитии общества, изменении содержания и характера деятельности человека.

Воспитательные:

- повышение общекультурного уровня учащихся;
- вооружение учащихся правильным методологическим подходом к познавательной и практической деятельности;
- выделение и раскрытие роли информационных технологий и компьютеров в развитии современного общества;
- привитие навыков сознательного и рационального использования компьютера в своей учебной, а затем и профессиональной деятельности;
- формирование эмоционально-ценностного отношения к миру, к себе;
- воспитание у учащихся стремления к овладению техникой исследования;
- воспитание трудолюбия, инициативности и настойчивости в преодолении трудностей.

Формы занятий: Основными, характерными при реализации данной программы формами являются комбинированные занятия. Занятия состоят из теоретической и практической частей, причём большее количество времени занимает практическая часть.

В теоретической части рассматриваются основные понятия языка программирования Python, основные алгоритмические конструкции. В практической части предлагаются

практические работы, направленные на отработку основных алгоритмических конструкций, на развитие логического мышления, на реализацию математических способностей учащихся в ходе составления программ. Практическая часть предполагает использование школьного компьютерного класса.

При проведении занятий традиционно используются три формы работы:

- демонстрационная, когда обучающиеся слушают объяснения педагога и наблюдают за демонстрационным экраном или экранами компьютеров на ученических рабочих местах;
- фронтальная, когда обучающиеся синхронно работают под управлением педагога;
- самостоятельная, когда обучающиеся выполняют индивидуальные задания в течение части занятия или нескольких занятий.

Технологии и формы обучения:

- теоретические занятия;
- практические занятия.

Режим занятий:

Занятия проводятся: 2 раза в неделю по 1 и 1,25 часа соответственно

Программа кружка предусматривает классно-урочную систему обучения.

Межпредметные связи

Знания, полученные при изучении курса «Программирование на языке Python », учащиеся могут использовать при создании собственных программ по определенной тематике, для решения задач из различных областей знаний – математике, физике, химии, биологии и др. Знания и умения, приобретенные в результате освоения данного курса, являются фундаментом для дальнейшего мастерства в области программирования.

МЕТОДИЧЕСКАЯ ЧАСТЬ

Основной тип занятий – практикум. Большинство заданий курса выполняется с помощью персонального компьютера и необходимых программных средств.

Занятия включают лекционную и практическую часть. Практическая часть курса реализуется через классно-урочную систему. Важной составляющей каждого урока является самостоятельная работа учащихся. Тема урока определяется приобретаемыми навыками. В

каждом уроке материал излагается следующим образом: повторение основных понятий и методов работы с ними, разбор новой темы, основные приемы работы (самостоятельное выполнение заданий для получения основных навыков работы), упражнения для самостоятельного выполнения.

Теоретическая и практическая части курса изучаются параллельно, чтобы сразу же закреплять теоретические вопросы на практике.

В ходе обучения учащимся периодически предлагаются короткие (5-10 мин) контрольные работы на проверку освоения изученных способов действий. Проводятся краткие срезовые работы (тесты, творческая работа) по определению уровня знаний учеников по данной теме. Выполнение контрольных способствует быстрой мобилизации и переключению внимания на осмысливание материала изучаемой темы. Кроме того, такая деятельность ведет к закреплению знаний и служит регулярным индикатором успешности образовательного процесса.

Регулярное повторение способствует закреплению изученного материала. Возвращение к ранее изученным темам и использование их при изучении новых тем способствуют устранению весьма распространенного недостатка – формализма в знаниях учащихся – и формируют научное мировоззрение учеников.

Планируемые результаты

В рамках данного курса учащиеся овладевают следующими знаниями, умениями и способами деятельности:

- знают роль программного обеспечения и его виды;
- у учащихся сформировано целостное представление об организации данных для эффективной алгоритмической обработки;
- знают основные алгоритмические конструкции и правила их записи, знакомы с основными способами организации данных;
- умеют составлять и записывать алгоритмы с использованием соответствующих алгоритмических конструкций;
- умеют распознавать необходимость применения той или иной алгоритмической конструкции при решении задачи;
- умеют организовывать данные для эффективной алгоритмической обработки;
- умеют разрабатывать алгоритмы и реализовывать их на языке программирования Python;
- умеют осуществлять отладку и тестирование программы.

Форма контроля за уровнем достижения учащихся

Предметом диагностики и контроля являются составленные алгоритмы и программы на языке программирования Python к предложенным задачам.

Оценка имеет различные способы выражения – устные суждения педагога, письменные качественные характеристики, систематизированные по заданным параметрам аналитические данные, в том числе и рейтинги. Оценке подлежит в первую очередь уровень достижения учеников минимально необходимых результатов, обозначенных в целях и задачах курса.

Качество знаний и умений ученика оценивается следующими характеристиками:

- знание основных алгоритмических конструкций;
- умение составить и записать алгоритм с использованием соответствующей алгоритмической конструкции;
- умение найти более эффективный способ решения задачи; □ умение тестировать программу.

В течение всего курса проводятся контрольные срезы и выставляются баллы за решенные задачи. По окончании курса **зачет** получают те учащиеся, у которых сумма баллов за учебный курс составляет не менее 60% от всей суммы баллов.

СОДЕРЖАНИЕ ПРОГРАММЫ

Часть 1. Теоретическая

1. Язык программирования Python

Текстовый редактор языка. Основы языка. Структура программы. Основные математические функции.

2. Ввод и вывод данных. Линейный алгоритм

Форматы вывода. Составление линейных алгоритмов.

3. Условный оператор

Условный оператор. Оператор выбора.

4. Алгоритмы с повторениями

Цикл с параметром FOR . Циклы While и Repeat. Вложенные циклы.

5. Массивы

Понятие массива. Двумерные массивы.

6. Графика в Python Графика.

Часть 2. Практическая

1. Язык программирования Python. Текстовый редактор языка.
2. Структура программы. Разделы описания.
3. Основные математические функции. Моя первая программа.
4. Ввод и вывод данных. Форматы вывода.
5. Составление линейных алгоритмов. Комментарии в программе.
6. Составление линейных алгоритмов с использованием арифметических операций.
7. Составление линейных алгоритмов с использованием основных функций.
8. Условный оператор. Структура условного оператора.
9. Условный оператор. Простые условия.
10. Условный оператор. Составные условия.
11. Операторные скобки.
12. Составной оператор.
13. Оператор выбора.
14. Алгоритмы с повторениями.
15. Цикл с параметром.
16. Цикл с предусловием.
17. Цикл с постусловием.
18. Вложенные циклы.
19. Понятие массива. Ввод и вывод элементов массива.
20. Задание массива в разделе констант.
21. Поиск в массиве элементов с заданными свойствами.
22. Поиск максимального (минимального) элемента массива.
23. Двумерные массивы: описание, ввод и вывод массивов по строкам.
24. Графика. Графические примитивы.
25. Структура графической программы. Прямые линии. Прямоугольники.
26. Графика. Окружность. Эллипс. Дуга. Сектор.
27. Графика. Закрашивание.
28. Построение графиков на экране.
29. Построение поверхностей.

Учебно-тематическое планирование элективного курса «Основы программирования»

№ п/п	Тема	Количество часов
1	Язык программирования Python	14
	Техника безопасности. Текстовый редактор языка. Основы языка. Структура программы. Основные математические функции.	5
	<i>Практическая работа №1.</i> Язык программирования Python . Текстовый редактор языка.	2,25
	<i>Практическая работа №2 .</i> Структура программы. Разделы описания.	2,25
	<i>Практическая работа №3.</i> Основные математические функции. Первая программа.	2,25
	<i>Практическая работа №4.</i> Ввод и вывод данных. Форматы вывода.	2,25
2	Ввод и вывод данных. Линейный алгоритм	8
	Форматы вывода. Составление линейных алгоритмов.	1,25
	<i>Практическая работа №5 .</i> Составление линейных алгоритмов. Комментарии в программе.	2,25
	<i>Практическая работа №6.</i> Составление линейных алгоритмов с использованием арифметических операций.	2,25
	<i>Практическая работа №7.</i> Составление линейных алгоритмов с использованием основных функций.	2,25
3	Условный оператор	14,75
	Условный оператор. Оператор выбора.	1,25
	<i>Практическая работа №8.</i> Условный оператор. Структура условного оператора.	2,25
	<i>Практическая работа №9.</i> Условный оператор. Простые условия.	2,25
	<i>Практическая работа №10.</i> Условный оператор. Составные условия.	2,25
	<i>Практическая работа №11.</i> Операторные скобки	2,25

	<i>Практическая работа №12. Составной оператор</i>	2,25
	<i>Практическая работа №13. Оператор выбора.</i>	2,25
4	Алгоритмы с повторениями	12,5
	Цикл с параметром FOR. Циклы While и Repeat. Вложенные циклы.	1,25
	<i>Практическая работа №14. Алгоритмы с повторениями.</i>	2,25
	<i>Практическая работа №15. Цикл с параметром.</i>	2,25
	<i>Практическая работа №16. Цикл с предусловием</i>	2,25
	<i>Практическая работа №17. Цикл с постусловием</i>	2,25
	<i>Практическая работа №18. Вложенные циклы.</i>	2,25
5	Массивы	12,5
	Понятие массива. Двумерные массивы.	1,25
	<i>Практическая работа №19. Понятие массива. Ввод и вывод элементов массива.</i>	2,25
	<i>Практическая работа №20. Задание массива в разделе констант.</i>	2,25
	<i>Практическая работа №21. Поиск в массиве элементов с заданными свойствами.</i>	2,25
	<i>Практическая работа №22. Поиск максимального (минимального) элемента массива.</i>	2,25
	<i>Практическая работа №23. Двумерные массивы: описание, ввод и вывод массивов по строкам.</i>	2,25
6	Графика в Python	14,75
	Графика.	1,25
	<i>Практическая работа №24. Графика. Графические примитивы.</i>	2,25
	<i>Практическая работа №25. Структура графической программы. Прямые линии. Прямоугольники.</i>	2,25
	<i>Практическая работа №26. Графика. Окружность. Эллипс. Дуга. Сектор</i>	2,25
	<i>Практическая работа №27. Графика. Закрашивание</i>	2,25

<i>Практическая работа №28. Построение графиков на экране.</i>	2,25
<i>Практическая работа №29. Построение поверхностей.</i>	2,25
ИТОГО:	76,5