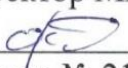


Муниципальное бюджетное общеобразовательное учреждение
«Средняя общеобразовательная школа №6»

РАССМОТРЕНО И ПРИНЯТО Решением педагогического совета, Протокол № 1 от 30.08.2022 г.	УТВЕРЖДЕНО Директор МБОУ СОШ №6  О.Б. Жидкова Приказ № 212/от 31.08.2022г.
--	---



Дополнительная общеобразовательная общеразвивающая программа

«Основы программирования на Python»

Возраст: 12-16 лет

Составитель: Усова И.А.
учитель информатики
кол-во часов в неделю: 2 ч
всего в год: 68 ч

г. Верхняя Салда

2022 год

1. Пояснительная записка

Настоящая дополнительная общеобразовательная общеразвивающая программа «Основы программирования на Python» (далее программа) имеет техническую направленность и ориентирована на развитие творческих способностей и технических умений обучающихся, организацию проектно-исследовательской деятельности, профессионального самоопределения.

Программа разработана в соответствии с основными нормативными документами:

1) Федеральный закон «Об образовании в Российской Федерации» (от 29.12.2012г №273-ФЗ);

2) Санитарные правила СП 2.4.3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (утверждено постановлением Главного государственного санитарного врача РФ от 28.09.2020 № 28);

3) Приказ Министерства образования и науки РФ от 9 января 2014 г. №2 «Об утверждении Порядка применения организациями, осуществляющими образовательную деятельность, электронного обучения, дистанционных образовательных технологий при реализации образовательных программ»;

4) Приказ Министерства просвещения РФ от 9 ноября 2018 г. № 196 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».

Актуальность программы продиктована развитием современного информационного общества, широким внедрением информационных технологий в образовательные процессы и обычную жизнь каждого человека. Реализация программы способствует развитию мотивации к получению новых знаний, возникновению интереса к программированию как к инструменту самовыражения в творчестве, а также поможет в повышении самооценки и самоопределении в профессиональной направленности личности. Таким образом, реализация программы позволит обучающимся получить необходимый

объем знаний о возможностях использования языка программирования Python, создать положительную мотивацию, способствующую развитию личностных, метапредметных и предметных умений обучающихся. Кроме того, реализация Программы даст возможность для создания условий для социального, культурного и профессионального самоопределения, творческой самореализации личности ребенка с помощью реализации мероприятий технической направленности, а также овладение ими soft и hard компетенциями.

Педагогическая целесообразность. Педагогическая целесообразность заключается в использовании игрового, визуального подхода для вовлечения детей в активность, чтобы они оставались вовлеченными в процессе обучения. Python не только язык программирования, но еще и интерактивная среда, где результаты действий визуализированы, что делает работу с программой понятной. А также наличие графического модуля, что позволяет визуализации работы ученика, что делает программу практически значимой для современного школьника.

Отличительные особенности программы. Отличительной особенностью программы является ее направленность на развитие у обучающихся гибких навыков: критическое мышление; креативность; коммуникация; планирование своих действий.

Возраст обучающихся: 12-16 лет

Этому возрасту свойственно чувство взрослости: потребность равноправия, уважения и самостоятельности, требование серьезного, доверительного отношения со стороны взрослых. Происходит формирование зрелых форм учебной мотивации, при которой учение приобретает личностный смысл. Приобретается опыт совместного действия в сообществе сверстников и значимых взрослых, объединенных общей деятельностью.

Количество обучающихся в группе: 5-10 человек

Объем программы: 128 ч.

Срок реализации программы: 2 года.

Режим занятий: 2 академических часа в неделю.

Длительность занятия: 40 минут

Форма обучения: очная

Формы организации учебных занятий:

- урок-получение новых знаний
- урок-консультация;
- практикум;
- урок-проект.

1. Цель и задачи программы

Цель - обучение программированию через создание творческих проектов, развитие личности ребенка, способного к творческому самовыражению, обладающего технической культурой, аналитическим мышлением, навыками и умениями программирования, умеющего работать в коллективе, способного применять полученные знания при решении бытовых и учебных задач.

Задачи:

Обучающие:

- обучить навыками составления алгоритмов;
- познакомить с понятиями «объект», «событие», «управление», «обработка событий»;
- изучить принципы работы основных алгоритмических конструкций;
- сформировать представление о профессии «программист»;
- сформировать навыки разработки, тестирования и отладки несложных программ;
- познакомить с понятием проекта и алгоритмом его разработки;
- сформировать навыки разработки проектов: интерактивных историй, интерактивных игр, мультфильмов, интерактивных презентаций.

Развивающие:

- раскрыть критический, системный, алгоритмический и творческий потенциал мышления;
- развить внимание, память, наблюдательность;

- развить познавательный интерес;
- развить умение работать с компьютерными программами и дополнительными источниками информации;

- развить навыки планирования проекта и умение работать в группе.

Воспитательные:

- сформировать культуру работы коллективной проектной деятельности при реализации общих проектов;
- способствовать развитию интереса к изучению профессий, связанных с программированием;
- способствовать формированию установки на позитивную социальную деятельность в информационном обществе, на недопустимость действий, нарушающих правовые, этнические нормы общения в информационном пространстве.

2. Учебно-тематический план

№ п/п	Наименование тем	Количество часов			Формы аттестации/контроля
		Всего	Теория	Практика	
Инструктаж по технике безопасности в компьютерном классе		2	2		Опрос
1.	Раздел 1. Знакомство с языком Python	6	2	4	Тестирование, решение практических задач
1.1	Общие сведения о языке Практическая работа: Установка программы Python	2	1	1	
1.2.	Режимы работы Практическая работа: Режимы работы с Python	2	1	1	
1.3	Тест № 1. Знакомство с языком Python	2		2	
2.	Раздел 2. Переменные и выражения	12	4	8	Тестирование, решение практических задач
2.1	Переменные Практическая работа: Работа со справочной системой	2	1	2	
2.2	Практическая работа: Переменные	2		2	
2.3	Арифметические выражения Практическая работа: Выражения	2	1	1	
2.4	Ввод и вывод	1	1		
2.5	Задачи на элементарные действия с числами	2	1	1	

	Практическая работа 2.5. Задачи на элементарные действия с числами				
2.6	Тест № 2. Выражения и операции.	1		1	
3	Раздел 3. Условные предложения	27	6	21	Тестирование, Решение практических задач
3.1	Логические выражения и операторы. Практическая работа: Логические выражения	2	1	1	
3.2	Условный оператор Практическая работа: «Условный оператор»	4	1	3	
3.3	Множественное ветвление Практическая работа: Множественное ветвление	4	1	3	
3.4	Реализация ветвления в языке Python. Практическая работа: «Условные операторы»	2	1	1	
3.5	Самостоятельная работа № 1 по теме «Условные операторы».	10	2	8	
3.6	Зачетная работа № 1. "Составление программ с ветвлением".	4		4	
3.7	Тест № 3. "Условные операторы".	1		1	
4	Раздел 4. Циклы	41	9	32	Тестирование, решение практических задач, творческая работа
4.1.	Оператор цикла с условием Практическая работа "Числа Фибоначчи"	4	1	3	
4.2.	Оператор цикла for Практическая работа Решение задачи с циклом for.	4	1	3	
4.3.	Вложенные циклы Практическая работа: Реализация циклических алгоритмов	4	1	3	
4.4.	Случайные числа Практическая работа: Случайные числа	4	1	3	
4.5.	Примеры решения задач с циклом Практическая работа: Решение задач с циклом.	6	1	5	
4.6	Самостоятельная работа № 2 "Составление программ с циклом"	10	2	8	
4.7	Тест № 4. Циклы	1		1	
4.8	Творческая работа № 1. "Циклы"	8	2	6	
5	Раздел 5. Функции	19	6	13	Тестирование, решение практических задач
5.1.	Создание функций Практическая работа Создание функций	2	1	1	
5.2.	Локальные переменные Практическая работа Локальные переменные	2	1	1	

5.3.	Примеры решения задач с использованием функций Практическая работа Решение задач с использованием функций	2	1	1	
5.4	Самостоятельная работа № 3 по теме "Функции"	10	2	8	
5.5	Рекурсивные функции Практическая работа 5.4. Рекурсивные функции	2	1	1	
5.6	Тест № 5. Функции	1		1	
6	Раздел 6. Строки - последовательности символов	10	3	7	
6.1.	Строки Практическая работа: Строки	2	1	1	Решение практически х задач
6.2.	Срезы строк	2	1	1	
6.3	Примеры решения задач со строками Практическая работа: Решение задач со строками.	6	1	5	
7	Раздел 7. Сложные типы данных	12	4	8	
7.1.	Списки Практическая работа: Списки	2	1	1	Тестирование, Решение практически х задач
7.2	Срезы списков	1	1		
7.3	Списки: примеры решения задач Практическая работа 7.2. Решение задач со списками	6	1	5	
7.4	Матрицы	2	1	1	
7.5	Тест № 7. Списки	1		1	
	ВСЕГО	128	36	92	

3. Содержание учебного плана

Вводное занятие (2 часа).

Теория: Знакомство с планом работы объединения, инструктаж по ТБ.

Форма аттестации: Опрос.

Раздел 1. Знакомство с языком Python (6 часов)

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

Учащиеся должны знать / понимать:

- понятие программы;
- структура программы на Python;
- режимы работы с Python.

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;

- написать комментарии в программе.

Раздел 2. Переменные и выражения (12 часов)

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

Выражения. Операции. Порядок выполнения операций. Математические функции.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. 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.

Учащиеся должны знать / понимать:

- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;

Раздел 3. Условные предложения (27 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".

Учащиеся должны знать / понимать:

- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- Учащиеся должны уметь:
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов.

Раздел 4. Циклы (41 час)

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"

Учащиеся должны знать / понимать:

- циклы с условием и их виды;
- правила записи циклов условием;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.

Учащиеся должны уметь:

- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использовать цикл с параметром для решения поставленной задачи.

Раздел 5. Функции (19 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

Учащиеся должны знать / понимать:

- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.

Учащиеся должны уметь:

- создавать и использовать функции;

- использовать механизм параметров для передачи значений.

Раздел 6. Строки - последовательности символов (10 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.

Учащиеся должны знать / понимать:

- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками.

Учащиеся должны уметь:

- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке.

Раздел 7. Сложные типы данных (12 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.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками.

Тест № 7. Списки

Учащиеся должны знать / понимать:

- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;

Учащиеся должны уметь:

- описывать списки;
- вводить элементы списка;
- выводить элементы списка;
- выполнять поиск элемента в списке, поиск минимума и максимума, нахождение суммы элементов списка;
- использовать вложенные списки;
- приводить примеры использования вложенных списков (матриц).

5. Планируемые результаты освоения программы

Планируемые результаты курса

В рамках курса «Программирование на языке Python» 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 «Основы программирования на языке Python» 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.

6. Комплекс организационно-педагогических условий

Календарный учебный график

Работа по программе осуществляется с 15 сентября по 25 мая. Предусмотрены каникулы.

Условия реализации программы

Материально-техническое обеспечение:

- столы для компьютера;
- компьютерные стулья;
- шкафы для дидактических материалов, пособий;
- специальная и научно-популярная литература для педагога и учащихся;
- канцтовары;

Информационное обеспечение:

- персональный компьютер (на каждого участника);
- мультимедийный проектор;
- видеоматериалы разной тематики по программе;
- оргтехника;
- выход в сеть Internet;

Аппаратное обеспечение:

- Процессор не ниже Core2 Duo;
- Объем оперативной памяти не ниже 4 ГбDDR3;
- Дисковое пространство на менее 128 Гб;
- Монитор диагональю на мене 19”;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- IDE Jupyter PyCharm;
- Foxit Reader или другой просмотрщик PDF файлов;
- WinRAR;
- Пакет офисных программ;
- Adobe Photoshop или другой растровый графический редактор;
- Любой браузер для интернет серфинга.

Формы аттестации (контроля)

Входящий контроль осуществляется при комплектовании группы в начале учебного года. *Цель* – определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточный контроль осуществляется в конце I полугодия учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

Итоговый контроль осуществляется в конце учебного года.

Форма контроля: защита творческого проекта.

Отслеживание личностного развития учащихся осуществляется методом наблюдения, анкетирования. По итогам первого полугодия и по итогам года заполняется «Диагностическая карта», в которой проставляется уровень усвоения программы каждым учащимся объединения.

Оценочные материалы

Входящий контроль: *Тестирование (Приложение 2)*

Промежуточный контроль: *Тестирование, решение задач (Приложение 3)*

Итоговый контроль: *учащимся предлагается самостоятельно выбрать тему творческого проекта и на основании темы разработать программу, пояснительную записку, презентацию.*

Методическое обеспечение

Программа разработана на основе концептуальных положений Стратегии инновационного развития Российской Федерации на период до 2020 года.

Методы обучения:

1. Словесный: объяснение нового материала; рассказ обзорный для раскрытия новой темы; беседы с учащимися в процессе изучения темы.
2. Наглядный: применение демонстрационного материала, наглядных пособий, презентаций по теме.

3. Практический: индивидуальная и совместная продуктивная деятельность, выполнение учащимися определенных заданий, решение задач.

4. Интерактивный: создание специальных заданий, моделирующих реальную жизненную ситуацию, из которой учащимся предлагается найти выход.

Технологии:

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.

2. Технология коллективного взаимообучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения (КСО), «работа учащихся в парах сменного состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.

3. Игровая технология. Игровая форма в образовательном процессе создаётся при помощи игровых приёмов и ситуаций, выступающих как средство побуждения к деятельности. Способствует развитию творческих способностей, продуктивному сотрудничеству с другими учащимися. Приучает к коллективным действиям, принятию решений, учит руководить и подчиняться, стимулирует практические навыки, развивает воображение.

4. Элементы здоровьесберегающих технологий являются необходимым условием снижения утомляемости и перегрузки учащихся.

5. Проектная технология предлагает практические творческие задания, требующие от учащихся их применение для решения проблемных заданий, знания материала на данный исторический этап. Овладевая культурой проектирования, школьник приучается творчески мыслить, прогнозировать возможные варианты решения стоящих перед ним задач.

6. Информационно-коммуникационные технологии активизируют творческий потенциал учащихся; способствует развитию логики, внимания, речи, повышению качества знаний; формированию умения пользоваться информацией, выбирать из нее необходимое для принятия решения, работать со всеми видами информации, программным обеспечением, специальными программами и т.д.

7. Список литературы

Нормативно-правовые акты и документы

1. Федеральный закон «Об образовании в Российской Федерации» от 29.12.2012 № 273-ФЗ
2. Приказ Министерства образования и науки РФ от 29.08.2013 №1008 «Об утверждении порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
3. Письмо Министерства образования Российской Федерации от 18 июня 2003 №28-02-484/16 «Требования к содержанию и оформлению образовательных программ дополнительного образования детей»,
4. Постановление Главного государственного санитарного врача Российской Федерации от 4.07.2014 г. «Об утверждении СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей».
5. Стратегия инновационного развития Российской Федерации на период до 2020 года. Распоряжение правительства Российской Федерации от 08.12.2011 № 2227-р.
6. Концепция развития дополнительного образования детей. Распоряжение Правительства Российской Федерации от 24 апреля 2015 года № 729-р.
7. Стратегия развития воспитания в Российской Федерации на период до 2025 года». Распоряжение Правительства Российской Федерации от 29 мая 2015 года № 996-р.

Литература для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.

3. Официальный сайт программы [Электронный ресурс] – Режим доступа: <https://docs.python.org/>, свободный.

4. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.

Литература для родителей и учащихся

1. Сайт / справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.

2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.

3. Сайт / Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431>, свободный.

4. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.

Сайт проекта Open Book Project. Практические примеры на Python Криса Мейерса [Электронный ресурс] – Режим доступа: openbookproject.net, свободный.