

Муниципальное бюджетное общеобразовательное учреждение
"Саконская средняя школа"

Принята на заседании
педагогического совета
от 30 августа 2023 г.
Протокол № 11

УТВЕРЖДАЮ
Директор МБОУ
Саконской СШ
_____ О.А. Поселеннова
от 30 августа 2023 г.

Дополнительная общеобразовательная общеразвивающая программа
технической направленности
"Программирование на языке Python"
Возраст обучающихся: 13-15 лет
Срок реализации: 1 год

Автор – составитель: учитель
Чухнин Александр Алексеевич,

Оглавление

Раздел №1 Комплекс основных характеристик программы _____	3
1.1 Пояснительная записка _____	3
1.2 Цель и задачи программы _____	4
1.3 Содержание программы _____	5
1.4 Планируемые результаты _____	11
Раздел №2 Комплекс организационно-педагогических условий _____	12
2.1 Календарный учебный график _____	12
2.2 Условия реализации программы _____	13
2.3 Формы аттестации _____	13
2.4 Оценочные материалы _____	14
2.5 Методические материалы _____	21
2.6 Список литературы _____	23

Раздел №1 Комплекс основных характеристик программы

1.1 Пояснительная записка

Изменение взглядов на программирование как науку, его место в системе научного знания требует существенных изменений в содержании образовательного процесса. В связи с этим особую актуальность приобретают раскрытие личностных резервов обучающихся и создание соответствующей образовательной среды.

Общепедагогическая направленность занятий – гармонизация индивидуальных и социальных аспектов обучения в отношении к информационным технологиям. Умение составлять алгоритмы решения и навыки программирования являются элементами информационной компетенции – одной из ключевых компетенций современной школы. Умение находить решение, составлять алгоритм решения и реализовать его с помощью языков программирования — необходимое условие подготовки современных учащихся. Особая роль отводится широко представленной в курсе системе рефлексивных заданий. Освоение рефлексии направлено на осознание учащимися того важного обстоятельства, что наряду с разрабатываемыми ими продуктами в виде программ на компьютере рождается основополагающий образовательный продукт: освоенный инструментарий. Именно этот образовательный продукт станет базой для творческого самовыражения обучающихся в форме различных программ.

Никакая система задач, какой бы хорошей она ни была, никакие тренинги памяти, внимания и т. п. не дают того эффекта, который возникает в случае, если учащиеся осознают необходимость решения тех или иных задач, если у них появляется острая необходимость к преодолению интеллектуальных трудностей.

Содержание обучения, представленное в программе «Программирование на языке «Python», 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 обучающихся знаний.

Дополнительная общеобразовательная общеразвивающая программа «Программирование на языке Python» (далее ДООП) является программой технической направленности.

Актуальность. С развитием современных информационных технологий сегодня любой обучающийся под руководством опытного педагога может с лёгкостью научиться программировать.

Компьютеры и компьютерные системы – неотъемлемая часть жизни нашего общества. Научившись программировать, мы можем быть не только пользователями информационных технологий, но и активными их создателями.

Языки программирования можно сравнить с иностранными языками, овладеть ими может каждый. Учиться программировать очень интересно. Результат программирования очень часто виден сразу. Кроме того, создание компьютерных игр и обучающих программ способствует развитию логики и креативного мышления. Ещё одной значимой стороной обучения программированию является спрос на рынке труда на специалистов данного направления деятельности.

Отличительной особенностью дополнительной общеобразовательной общеразвивающей программы «**Программирование на языке Python**» является её направленность на формирование у обучающихся навыков поиска собственного решения поставленной задачи, составления алгоритма решения и его реализации с помощью средств программирования

Возраст детей. Программа рассчитана на детей с 13 до 15 лет. Принимаются все желающие.

Сроки реализации. Программа рассчитана на 1 год обучения, 72 часа в год.

Форма обучения – очная.

Форма проведения занятий – аудиторная.

Форма организации занятий – всем составом объединения.

Режим занятий. Программа реализуется в течение всего учебного года, включая каникулярное время. Периодичность занятий: 1 раз в неделю по 2 часа.

1.2 Цель и задачи программы

Цель Программы: Способствовать формированию 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 Программы

Обучающие:

- обучить языку программирования Python и созданию программ на его основе;
- научить создавать прикладное программное обеспечение;
- расширять кругозор обучающихся в области программирования;
- научить дизайнерскому оформлению созданного ПО.

Развивающие:

- развивать память и внимание, познавательную и творческую активность;
- развивать творческие способности, эстетическое и эргономическое восприятие объектов труда;

- развивать профессиональное самоопределение обучающихся при обучении программированию;
- развивать логическое мышление.

Воспитательные:

- прививать интерес к активному творческому самовыражению, культуре труда;
- воспитывать упорство в достижении желаемого результата;
- воспитывать эстетический вкус;
- воспитывать чувство взаимопомощи, доверия, коллективизма.

1.3 Содержание программы

Учебный план

№	Название разделов	Год обучения		Кол-во часов
		1 полугодие (в том числе аттестация)	2 полугодие (в том числе аттестация)	
1	Введение в Python	7	-	7
2	Алгоритмические конструкции	11		11
3	Строки	10		10
4	Функции	6(1)	4	10(1)
5	Списки и кортежи		10	10
6	Словари и множества		4	4
7	Объектно-ориентированное программирование		18	18
8	Аттестация			
9	Итоговое занятие		1(1)	1(1)
Всего часов:		34(1)	38(1)	72(2)

Учебно-тематический план

№ п/п	Наименование раздела, темы	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Введение в Python	7	3	4	
1.1.	Язык Python. Среда программирования. Особенности ввода-	2	1	1	

	вывода				
1.2.	Типы данных, операции. Оператор присваивания	2	1	1	
1.3.	Числа. Стандартные операции	3	1	2	
2.	Алгоритмические конструкции	11	3	8	
2.1.	Условный оператор	1	1	-	
2.2.	Цикл while. Теория чисел	5	1	4	
2.3.	Цикл for	5	1	4	
3.	Строки	10	4	6	
3.1.	Литералы строк	2	1	1	
3.2.	Срезы строк	2	1	1	
3.3.	Методы строк	6	2	4	
4.	Функции	10	4	6	Защита проекта
4.1.	Парадигма и преимущества структурного программирования	1	1	-	
4.2.	Граф и стек вызовов функций. Области видимости переменных	2	1	1	
4.3.	Прямая рекурсия	5	1	4	
4.4.	Косвенная рекурсия	2	1	1	
5.	Списки и кортежи	10	4	6	
5.1.	Списки и кортежи в Python. Сходства и различия	1	1	-	
5.2.	Операции со списками	5	1	4	
5.3.	Срезы списков	4	1	1	
5.4.	Матрицы. Операции над матрицами.	2	1	1	
6	Словари и множества	4	2	2	
6.1.	Словари	2	1	1	
6.2.	Множества	2	1	1	
7.	Объектно-ориентированное программирование	18	2	16	
7.1.	Классы в Python	1	1	-	
7.2.	Разработка собственного класса	3	1	2	
7.3.	Разработка и программирование собственного проекта	10	-	10	
7.4	Программист: кто это и чем занимается?	4		4	
8.	Защита проекта	1	-	1	Защита проекта
9.	Итоговое занятие	1		1	
	Всего	72	22	50	

Содержание учебного плана

1. Введение в Python

Основные понятия: трансляция, интерпретация, компиляция, синтаксис, семантика, прагматика, переменная, динамическая типизация, служебные слова, идентификаторы, простые типы данных, приоритеты операций, литералы чисел, операция присваивания, PEP 8.

1.1. Язык Python. Среда программирования. Особенности ввода-вывода

Теория. Язык программирования Python. Достоинства и недостатки. Области применения. Интерактивный режим работы программы.

Практика. Установка языка программирования Python 3.8 и среды программирования WingIDE 101. Регистрация на Интернет-ресурсах.

1.2. Типы данных, операции. Оператор присваивания

Теория. Ввод и вывод числовой информации.

Практика. Тренировочное задание на ввод и вывод числовой информации.

1.3. Числа. Стандартные операции

Теория. Стандартные операции с целыми и действительными числами. Стилль программирования Python (PEP 8).

Практика. Решение простых задач в интерактивном режиме.

2. Алгоритмические инструкции

Основные понятия: логический тип данных, логические операции (and, or, not, A), условный оператор, условное и альтернативное исполнение алгоритма, операторы сравнения, вложенность операторов, оператор цикла, переменная-флаг, генерация псевдослучайной последовательности, инструкции break, continue и pass.

2.1. Условный оператор

Теория. Условная и циклическая инструкции. Каскадность и вложенность алгоритмических инструкций.

2.2. Цикл while

Теория. Теория чисел. Фильтрация потока чисел. НОД иНОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики.

2.3. Цикл for

Теория. Теория чисел. Фильтрация потока чисел. НОД иНОК. Проверка числа на простоту. Алгоритм Евклида. Нахождение максимума и минимума.

Практика. Решение задач на анализ чисел потока и целочисленной арифметики повышенной трудности.

3. Строки

Основные понятия: символ, строка, литерал, таблицы кодовASCII,UTF-8, отладка кода, неизменяемый объект, формат вывода строки, экранированные escape- последовательности, положительная и отрицательная нумерация символов в строке, срез, конкатенация, длина строки.

3.1. Литералы строк

Теория. Понятие«литералы строк».

Практика. Ввод-вывод строки. Решение задач на ввод строки, поиск подстроки.

3.2. Срезы строк

Теория. Форматирование строки.

Практика. Преобразование строки. Решение упражнений.

3.3. Методы строк

Теория. Методы работы со строкой.

Практика. Применение методов строки. Решение задач.

4. Функции

Основные понятия: подпрограмма, функция,процедура,рекурсия,глубина рекурсии, объявление, определение и вызов функции, возврат значений, глобальные и локальные переменные, передача параметров, работа с памятью, граф вызовов, стек вызовов, полиморфизм функций, утиная типизация, lambda-функции.

4.1. Парадигма и преимущества структурного программирования

Теория. Обзор парадигм программирования. Особенности применения языков программирования.

4.2. Граф и стек вызовов функций. Области видимости переменных

Теория. Применение стек и граф вызовов функций.

Практика. Выполнение тренировочных упражнений.

4.3. Прямая рекурсия

Теория. Понятие прямой рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений.

4.4. Косвенная рекурсия

Теория. Понятие косвенной рекурсии и ее применение.

Практика. Выполнение тренировочных упражнений.

Практические занятия к темам 4.2., 4.3., 4.4.: Нахождение суммы чисел. Вычисление степени. Числа Фибоначчи. Ханойские башни

Использование библиотеки математических функций. Решение задач повышенной трудности.

5. Списки и кортежи

Основные понятия: список, кортеж, элемент списка и кортежа, индекс, срез списка, матрица, многомерный список, сортировка, сложность алгоритма, устойчивость сортировки, квадратичная, быстрая, синхронная, поразрядная сортировки списка, случайное перемешивание.

5.1. Списки и кортежи в Python. Сходства и различия

Теория. Представление списка и кортежа в памяти компьютера, сходства и различия.

5.2. Операции со списками

Теория. Способы заполнения списка(с клавиатуры, из файла, случайным образом, по формуле).

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка.

5.3. Срезы списков

Теория. Методы работы со списком и кортежем. Методы сортировки списка.

Практика. Решение задач на ввод-вывод элементов одномерного и многомерного списка и кортежа, преобразование, поиск, замену, подсчет.

5.4. Матрицы. Операции над матрицами

Теория. Вычисление сложности алгоритма. Многомерные списки.

Практика. Решение задач повышенной трудности.

6. Словари и множества

Основные понятия: *словарь, множество, ключ, кодирование.*

6.1. Словари

Теория. Понятие словаря. Способы создания. Словарь, преимущества и недостатки, методы работы со словарем. Словари со смешанными значениями. Кодирование и декодирование текста.

Практика. Решение задач на заполнение, преобразование, поиск, замену, подсчет, вывод элементов словаря.

6.2. Множества

Теория. Понятие множества. Создание множеств. Множество, преимущества и недостатки, методы работы с множеством.

Практика. Решение задач повышенной трудности.

7. Объектно-ориентированное программирование (ООП)

Основные понятия: ООП, класс, метод INIT, экземпляр, наследование, полиморфизм, исключения, виджет, интерфейс, событие, техническое задание, проект, проектная деятельность, виды проектов.

7.1. Классы в Python

Теория. Понятия «класс», «метод INIT», «экземпляр», «наследование», «полиморфизм», «исключения», «виджет», «интерфейс», «интерфейс», «событие».

7.2. Разработка собственного класса

Теория. Принципы разработки собственного класса. Обработка и генерация исключений. Виджет, методы виджета. Графическая библиотека tkinter, класс Tk. Системные методы. События.

Практика. Создание собственного класса.

7.3. Разработка и программирование собственного проекта

Практика. Выбор вида и темы проекта. Составление технического задания. Программирование. Разработка технической документации и презентации проекта.

7.4. Программист: кто это и чем занимается?

Теория. Программист. Описание профессии. Плюсы профессии. Типы профессии.

8. Аттестация: защита проекта

9. Итоговое занятие

Планируемые результаты

Обучающиеся должны знать:

- принципы программирования на языке Python;
- основы дизайнерского оформления созданных программ;
- понятие программы;
- структура программы на Python.

Обучающиеся должны уметь:

- производить чтение и запись программ на языке Python;
- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python.
- работать над проектом в команде, эффективно распределять обязанности.
- представить и защитить свой проект;

2.2 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 Гб DDR3;
- Дисковое пространство не менее 128 Гб;
- Монитор диагональю не менее 19”;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- IDE JupyterLab/PyCharm;
- FoxitReader или другой просмотрщик PDF файлов;
- WinRAR;
- Пакет офисных программ;
- AdobePhotoshop или другой растровый графический редактор;
- Любой браузер для интернет серфинга.

2.3 Формы аттестации

В процессе изучения программы для оценки текущей работы используются следующие методы: педагогическое наблюдение, выполнение практических работ в тетрадях и на компьютерах, выполнение проектных заданий по определенной теме и защита своих работ перед группой, участие в конкурсах.

Форма подведения итогов реализации дополнительной общеобразовательной общеразвивающей программы – защита проекта. Это мероприятие является контрольным и служит показателем освоения детьми программы.

2.4 Оценочные материалы

Промежуточный контроль осуществляется в конце I полугодия и II полугодия.

Форма контроля: защита проекта.

Отслеживание личностного развития обучающихся осуществляется методом наблюдения, анкетирования.

Высокий уровень освоения программы	учебной и творческой деятельности, которая является содержанием программы; показывает широкие возможности практического применения в собственной творческой деятельности приобретенных знаний умений и навыков
Средний уровень освоения программы	Обучающийся демонстрирует достаточную заинтересованность в учебной и творческой деятельности, которая является содержанием программы; может применять на практике в собственной творческой деятельности приобретенные знания умения и навыки
Низкий уровень освоения программы	Обучающийся демонстрирует слабую заинтересованность в учебной и творческой деятельности, которая является содержанием программы; не стремится самостоятельно применять на практике в своей деятельности приобретенные знания умения и навыки

2.5 Методические материалы

Методы обучения и воспитания

Методы обучения:

Методы обучения, в основе которых лежит способ организации занятия:

- словесный (беседа, объяснение, рассказ);
- наглядный (показ видеоматериалов, иллюстраций);
- практический (тренировочные упражнения).

Методы, в основе которых лежит уровень деятельности детей:

- объяснительно-иллюстративные (методы обучения, при использовании которых, дети воспринимают и усваивают готовую информацию);
- репродуктивные методы обучения (обучающиеся воспроизводят полученные знания и освоенные способы деятельности);
- частично-поисковые методы обучения (участие детей в коллективном

поиске, решение поставленной задачи совместно с педагогом).

Методы воспитания:

- методы формирования сознания (методы убеждения);
- методы организации деятельности и формирования опыта поведения;
- методы стимулирования поведения и деятельности.

Формы организации учебного занятия

Основными формами проведения занятий являются: лекция, практическое занятие.

Педагогические технологии

На занятиях объединения используются следующие современные педагогические технологии:

Личностно – ориентированные технологии:

- технология полного усвоения знаний, основанная на общей установке: все учащиеся способны полностью усвоить необходимый материал при условии рациональной организации учебно-тренировочного процесса;
- технология разноуровневого обучения, в качестве теоретического обоснования используют теорию структуры личности, предполагают дифференцированное обучение с целью выполнения заданий по силам и продвижению каждого теннисиста вперед.

Здоровьесберегающие технологии:

- наличие эмоциональных разрядок на занятиях;
- использование упражнений для глаз;
- мобилизация внутренних сил (похвала, поддержка)

Целью здоровьесберегающих технологий должно выступать обеспечение условий физического, психологического, социального и духовного комфорта, способствующих сохранению и укреплению здоровья юных спортсменов, их продуктивной учебно – познавательной и практической деятельности, основанной на научной организации труда и культуре здорового образа жизни личности.

Алгоритм учебного занятия

Занятия строятся по следующему алгоритму.

1 этап: организационный.

Задача: подготовка детей к работе на занятии.

Содержание этапа: организация начала занятия, создание психологического настроения на учебную деятельность и активизация внимания.

2 этап: проверочный.

Задача: установление правильности и осознанности выполнения домашнего задания (если было), диагностика усвоения, выявление пробелов и их коррекция.

Содержание этапа: проверка усвоения знаний предыдущего занятия.

3 этап: подготовительный (подготовка к новому содержанию).

Задача: обеспечение мотивации и принятие детьми цели учебно-познавательной деятельности.

Содержание этапа: сообщение темы, цели учебного занятия и мотивация учебной деятельности детей.

4 этап: основной.

В качестве основного этапа выступают следующие:

1) Усвоение новых знаний и способов действий.

Задача: обеспечение восприятия, осмысления и первичного запоминания связей и отношений в объекте изучения.

Содержание этапа: при усвоении новых знаний используются задания и вопросы, которые активизируют познавательную деятельность детей.

2) Первичная проверка понимания.

Задача: установление правильности и осознанности усвоения нового учебного материала, выявление неверных представлений и их коррекция.

Содержание этапа: применяются пробные практические задания, которые сочетаются с объяснением соответствующих правил или их обоснованием.

3) Закрепление знаний и способов действий.

Задача: обеспечение усвоения новых знаний и способов действий.

Содержание этапа: применяют тренировочные упражнения, задания, которые выполняются самостоятельно детьми.

4) Обобщение и систематизация знаний.

Задача: формирование целостного представления знаний по теме.

Содержание этапа: распространенными способами работы являются беседа и практические задания.

5 этап: контрольный.

Задача: выявление качества и уровня овладения знаниями, их коррекция.

Содержание этапа: используются тестовые задания, виды устного и письменного опроса, вопросы и задания различного уровня сложности (репродуктивного, творческого, поисково-исследовательского).

6 этап: итоговый.

Задача: дать анализ и оценку успешности достижения цели и наметить перспективу последующей работы.

Содержание этапа: сообщаются ответы на следующие вопросы: как работали ребята на занятии, что нового узнали, какими умениями и навыками овладели? Поощрение ребят за учебную работу.

7 этап: рефлексивный.

Задача: мобилизация детей на самооценку.

Содержание этапа: оценивается работоспособность, психологическое состояние, результативность работы, содержание и полезность учебной работы.

8 этап: информационный

Задача: обеспечение понимания цели, содержания и способов выполнения домашнего задания, логики дальнейших занятий.

Содержание этапа: информация о домашнем задании (если необходимо),

инструктаж по его выполнению, определение перспективы следующих занятий.

Типы занятий

- занятие изучения нового материала;
- занятия применения и совершенствования полученных знаний;
- комбинированные занятия;
- контрольные занятия.

2.6 Список литературы

Нормативно-правовые документы

1. 1. Федеральный закон от 29 декабря 2012 года №273-ФЗ «Об образовании в Российской Федерации».
2. Концепция развития дополнительного образования детей до 30 года, утвержденная распоряжением Правительства Российской Федерации от 31 марта 2022 года №678-р).
3. Приказ Министерства просвещения Российской Федерации от 27.07.2022 г. №629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам».
4. Письмо Минобрнауки России от 11.12.2006 г. № 06-1844 «О примерных требованиях к программам дополнительного образования детей».
5. Постановление Главного государственного санитарного врача Российской Федерации от 28 сентября 2020 г. N 28 г. Москва "Об утверждении санитарных правил С.П.2.4. 3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи".
6. Постановление Главного государственного санитарного врача Российской Федерации от 28 января 2021 г. N 2 г. Москва "Об утверждении санитарных правил и норм СанПиН 1.2.3685-21 "Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания".
7. Устав муниципального бюджетного общеобразовательного учреждения "Саконская средняя школа".

Список литературы, используемой педагогом в работе

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.

Список литературы для обучающихся

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