


МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«УВИНСКАЯ СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА №4»

ПРИНЯТО
На заседании педагогического совета
Протокол № 9 от 30.08.2023

УТВЕРЖДЕНО
Приказ директора МОУ «Увинская СОШ №4»
№ 18/к от 30 августа 2023
Директор МОУ «Увинская СОШ №4» Е.С. Аргандеева



**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

«Программирование на Python»

Разработчики: Антонов Е.В., Ившина Е.А.,
педагоги дополнительного образования

пос. Ува

2023-2024 учебный год

I. КОМПЛЕКС ОСНОВНЫХ ХАРАКТЕРИСТИК ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ ПРОГРАММЫ

1. ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» разработана в соответствии с нормативными документами и не противоречит законодательству РФ.

1.1. Направленность программы

Настоящая программа имеет *техническую направленность*, рассчитана на учащихся 7-11 классов и предназначена для развития навыков алгоритмического мышления. Курс последовательно и целенаправленно вовлекает учащегося в процесс самостоятельного и осмысленного составления законченных алгоритмов и программ, вырабатывает необходимые составляющие алгоритмической и программистской грамотности.

1.2. Актуальность программы

Язык программирования Python на сегодняшний день является наиболее популярной высокоуровневой средой международного олимпиадного программирования. Python –мощный и простой для изучения язык программирования. В нем представлены проработанные высокоуровневые структуры данных и простой, но эффективный подход к объектно-ориентированному программированию.

Python – серьёзный язык программирования. В то же время учащиеся в полной мере могут раскрыть свои творческие таланты, так как с его помощью можно легко создавать игры и другие приложения.

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.

1.3. Отличительные особенности программы

1. Проектный подход. В процессе обучения происходит воспитание культуры проектной деятельности, раскрываются и осваиваются основные шаги по разработке и созданию проекта.

2. Межпредметность. В курсе прослеживается тесная взаимосвязь с математикой, физикой, географией, русским языком, музыкой и другими предметами школьного цикла. Знания, полученные на других предметах, логичным образом могут быть использованы при разработке проектов.

3. Вариативность. Учащиеся с достаточной степенью свободы и самостоятельности могут выбирать темы проектов.

4. Коммуникация. В курсе предусмотрена работа в командах, парах, использование возможностей сетевого сообщества для взаимодействия. Обязательное условие - публичная презентация и защита проектов.

1.4. Адресат программы

Программа рассчитана на обучающихся 13 – 16 классов.

1.5. Объем программы

Срок реализации программы – 1 год (68 часов).

1.6. Организация образовательного процесса

Режим занятий: 2 занятия в неделю, общее количество часов в год – 68.

Продолжительность занятия – 40 минут.

Распределение учебного времени занятия определяется в соответствии с возрастом детей и требованиями СанПиН.

При организации образовательного процесса используются фронтальные, групповые и индивидуальные формы работы.

2. ЦЕЛЬ И ЗАДАЧИ ПРОГРАММЫ

Цель: 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:

Обучающие:

- обучение основам алгоритмизации и программирования, приобщении к проектно-творческой деятельности;

- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных;
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Воспитательные:

- воспитывать интерес к занятиям информатикой;
- воспитывать культуру общения между учащимися;
- воспитывать культуру безопасного труда при работе за компьютером;
- воспитывать культуру работы в глобальной сети;
- воспитание целеустремленности и результативности в процессе решении учебных задач;

Развивающие:

- развивать познавательный интерес школьников;
- развивать творческое воображение, математическое и образное мышление учащихся;
- развивать умение работать с компьютерными программами и дополнительными источниками информации;
- развивать навыки планирования проекта, умение работать в группе

3. СОДЕРЖАНИЕ ПРОГРАММЫ

3.1. Учебно-тематическое планирование

№ п/п	Название разделов, тем	Количество часов			Формы и методы организации занятий	Формы аттестации (контроля)
		Всего	Теор	Практ		
1.	Введение. Инструктаж по ТБ. Знакомство с	1	0,5	0,5	Инструктаж Демонстрация /показ/	Опрос

	правилами работы за компьютером.				Работа за компьютером	
2.	Основы языка программирования Python	30	10	20		
2.1.	Введение, типы данных, синтаксис	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.2.	Арифметические операции	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.3.	Логика. Логические операторы	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.4.	Условный оператор. Сложные условия	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.5.	Цикл for	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.6.	Цикл while	3	1	2	Объяснение нового материала Демонстрация /показ/	Тестирование Решение практических

					Практическая работа	задач
2.7.	Строковый тип данных	3	1	2	Объяснение нового Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.8.	Списки и индексация в них	6	2	4	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
2.9.	Функции, лямбды	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
3.	Алгоритмы	6	2	4		
3.1	Арифметические алгоритмы	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
3.2	Сортировки. Понятие асимптотической сложности	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.	Структуры данных	12	4	8		
4.1.	Многомерные массивы	3	1	2	Объяснение нового материала	Тестирование Решение

					Демонстрация /показ/ Практическая работа	практических задач
4.2.	Кортежи, множества, словари	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.3.	Работа со словарями	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
4.4.	Стек, очередь, дек	3	1	2	Объяснение нового материала Демонстрация /показ/ Практическая работа	Тестирование Решение практических задач
5.	Разработка и защита проекта (кейса)	19	1	18		
5.1.	Алгоритм создания проекта.	2	1	1	Объяснение нового материала Демонстрация /показ	План проекта
5.2.	Подбор материалов для проекта	3		3	Объяснение нового материала Демонстрация /показ	Библиотека материалов
5.3.	Создание программы с использованием подготовленных материалов	9		9	Демонстрация /показ/ Практическая работа	Проект
5.4.	Тестирование и отладка проекта.	2		2	Практическая работа	Проект

5.5.	Подготовка к защите проекта.	1		1	Самостоятельная работа, консультации с преподавателем	План защиты проекта.
5.6.	Защита проекта.	2		2	Демонстрация и защита собственного проекта Публичное выступление	Творческий проект
	Итого	68	17,5	50,5		

3.2. Содержание учебно-тематического плана

1. Введение. Инструктаж по ТБ. Знакомство с правилами работы за компьютером. (1 час)

Теория: Знакомство с программой творческого объединения. Правила поведения на занятии. Техника безопасности при работе на компьютере.

Практика: Практическая работа за компьютером.

2. Основы языка программирования Python (30 часов)

2.1. Введение, типы данных, синтаксис.

Теория: О языке Python. Установка Python и сред разработки. Синтаксис языка Python. Переменные в Python. Типы данных и приоритеты операций в языке Python. Преобразование типов в Python. Ввод-вывод в Python. Где применяется Python.

Практика: Решение задач.

2.2. Арифметические операции.

Теория: Целочисленная арифметика. Целочисленный тип (int) в Python.

Практика: Решение задач.

2.3. Логика. Логические операторы.

Теория: Условная инструкция в Python. Ветвление в программе. Логический тип (bool).

Практика: Решение задач.

2.4. Условный оператор. Сложные условия.

Теория: Условный оператор: задачи на шахматной доске. Построение сложных условий. Разные задачи на условный оператор.

Практика: Решение задач.

2.5. Цикл for.

Теория: Цикл for в Python. Понятие об итераторе. Функция range(). Решение задач с циклом for.

Практика: Решение задач.

2.6. Цикл while.

Теория: Цикл while в Python. Инструкция управления циклом в Python. Отличия от цикла for. Применимость в каждом конкретном случае.

Практика: Решение задач.

2.7. Строковый тип данных.

Теория: Строки в Python. Срезы строк в Python. Методы строк в Python. Задачи на обработку строк.

Практика: Решение задач.

2.8. Списки и индексация в них.

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.

Практика: Решение задач.

2.9. Функции, лямбды.

Теория: Функции в Python. Локальные и глобальные переменные. Парадигма структурного программирования. Математические функции в Python. Лямбда-функции в Python.

Практика: Решение задач.

3. Алгоритмы (6 часов)

3.1. Арифметические алгоритмы

Теория: Что такое алгоритм. Различные формы записи алгоритмов (нумерованный список, таблица, блок-схема). Примеры линейных алгоритмов, алгоритмов с ветвлениями и повторениями (в повседневной жизни, в литературных произведениях, на уроках математики. Решето Эратосфена. Факторизация. Проверка на простоту.

Практика: Решение задач.

3.2. Сортировки. Понятие асимптотической сложности.

Теория: Асимптотическая сложность алгоритма. Сортировка подсчётом. Поразрядная сортировка. Быстрая сортировка Хоара. Сортировка слиянием.

Практика: Решение задач.

4. Структуры данных (12 часов)

4.1. Многомерные массивы

Теория: Множества в Python. Двумерные и многомерные списки в Python.

Практика: Решение задач.

4.2. Кортежи, множества, словари

Теория: Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения. Множества в языке Python. Множества. Множественный тип данных. Описание множеств. Операции, допустимые над множествами: объединение, пересечение, разность, включение. Оператор определения принадлежности элемента множеству. Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Практика: Решение задач.

4.3. Работа со словарями

Теория: Генераторы списков.

Практика: Решение задач.

4.4. Стек, очередь, дек.

Теория: Стек, очередь, дек.

Практика: Решение задач.

5. Разработка и защита проекта (кейса). (19 часов)

5.1. Алгоритм создания проекта.

Теория: жизненный цикл проекта.

Практика: разработка плана проекта.

5.2. Подбор материалов для проекта.

Теория: Поиск информации в Интернете.

Практика: Поиск информации по теме проекта.

5.3. Создание программы с использованием подготовленных материалов.

Практика: Создание программного кода. Разработка собственного проекта, его программирование, дизайн, оформление.

5.4. Тестирование и отладка проекта.

Практика: Тестирование и отладка проекта.

5.5. Подготовка к защите проекта.

Практика: подготовка к публичному выступлению

5.6. Защита проекта.

Практика: Публичное выступление. Защита проекта.

4. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ ДООП

При освоении программы отслеживаются три вида результатов: предметный, метапредметный и личностный, что позволяет определить динамическую картину творческого развития обучающихся.

Личностные

- сформированность мировоззрения, соответствующего современному уровню развития науки и техники;
- потребность в самореализации в творческой деятельности, желании учиться;
- умение осуществлять совместную информационную деятельность, в частности, при выполнении учебных проектов;
- осознанный выбор будущей профессии и возможностей реализации собственных жизненных планов; отношение к профессиональной деятельности как возможности участия в решении личных, общественных, государственных, общенациональных проблем.

Метапредметные

- умение самостоятельно выделять и формулировать познавательные цели;
- умение применять методы моделирования;
- умение прогнозировать результат деятельности и его характеристики, вносить необходимые коррективы в план по ходу его выполнения;

- приобретение навыков самостоятельного создания способов решения проблем творческого и поискового характера;
- умение использовать средства ИКТ в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;
- готовность и способность к самостоятельной информационно-познавательной деятельности, включая умение ориентироваться в различных источниках информации, критически оценивать и интерпретировать информацию, получаемую из различных источников;
- умение продуктивно общаться и взаимодействовать в процессе совместной деятельности, учитывать позиции других участников деятельности, эффективно разрешать конфликты.

Предметные

- умение интерпретировать сообщения с позиции их смысла, синтаксиса, ценности;
- умение планировать действия, необходимые для достижения заданной цели (связанной с преобразованием информации) с помощью фиксированного набора средств, способность реализовать эти действия с использованием языка программирования, оценивать результаты работы;
- приобретение опыта планирования учебного сотрудничества с учителем и сверстниками: определение цели, функции участников, способов взаимодействия;
- овладение понятием сложности алгоритма, знание основных алгоритмов обработки числовой и текстовой информации, алгоритмов поиска и сортировки;
- овладение навыками алгоритмического мышления и пониманием необходимости формального описания алгоритмов;
- умение реализовывать алгоритмы в виде программ и программных систем; владение элементарными навыками документирования программ.
- умение сопоставлять математические модели задач и их компьютерные аналоги, анализировать полученные результаты с точки зрения соответствия объекту и целям моделирования;
- умение анализировать разные способы записи алгоритмов с позиции того, что они являются информационными моделями;

- владение универсальным языком программирования высокого уровня, представлениями о базовых типах данных и структурах данных; умением использовать основные управляющие конструкции языка; умением производить отладку программ с помощью встроенного отладчика.

II. КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ

1. КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК /с 01.09.2020г. по 25.05.2021г./

Месяц	Сроки изучения учебного материала/нед	№ группы/год обучения/			Аттестация обучающихся	Каникулярный период
		Количество часов				
		№ 1/1		-		
Сентябрь	4-9	1				
	11-16	1				
	18-23	1				
	25-30	1				
	2-7	1				
Октябрь	9-14	1				
	16-21	1				
	23-28	1				
	30-4	Самостоятельная работа				
Ноябрь	7-11	1				
	13-18	1				
	20-25	1				
	27-1	1				
Декабрь	4-9	1				
	11-16	1				
	18-22	1				
	25-30	1				
	1-7	1				
Январь	8-13	Самостоятельная				

		работа				
	15-20	1				
	22-27	1				
	29-3	1				
Февраль	5-10	1				
	7-11	1				
	12-17	1				
	19-24	1				
Март	26-1	1				
	4-9	1				
	11-16	Самостоятельная работа				
	18-23	1				
Апрель	25-30	1				
	1-6	1				
	8-13	1				
	15-20	1				
Май	22-27	1				
	29-4	1				
	6-11	1			Защита проектов	
	13-18					
Всего учебных недель						
Всего часов по программе		34	34			

2. УСЛОВИЯ РЕАЛИЗАЦИИ ПРОГРАММЫ

2.1. Материально-техническое обеспечение

Для эффективности образовательного процесса необходимы:

Техническое оборудование:

Компьютер

Периферийные устройства: Сканер, принтер, проектор.

Локальная сеть, выход в Интернет.

Операционная система: Windows

Среда программирования РУНТОН.

2.2. Кадровое обеспечение

Программа реализуется педагогом дополнительного образования, имеющим профессиональное образование по направленности программы.

3. ФОРМЫ АТТЕСТАЦИИ/КОНТРОЛЯ

Система мониторинга достижения детьми планируемых результатов освоения дополнительной общеобразовательной общеразвивающей программы обеспечивает комплексный подход к оценке текущих и промежуточных результатов программы и позволяет осуществить оценку динамики достижения обучающихся, а также включает описание объекта, форм, периодичность и содержание мониторинга. Отслеживание результатов осуществляется через входной, текущий, промежуточный и итоговый контроль.

Время проведения	Цель проведения	Формы контроля
Входной контроль		
В начале учебного года	Определение уровня развития детей, их знания ИКТ.	беседа
Текущий контроль		

В течение учебного года	<p>Определение степени усвоения обучающимися учебного материала.</p> <p>Определение готовности детей к восприятию нового материала. Выявление детей, отстающих и опережающих обучение. Подбор наиболее эффективных методов и средств обучения.</p>	<p>Тестирование</p> <p>Решение практических задач</p> <p>Участие в конкурсах, олимпиадах</p>
Промежуточный контроль		
В конце месяца, полугодия	<p>Определение степени усвоения обучающимися учебного материала.</p> <p>Определение результатов обучения.</p>	Практические и творческие задания
Итоговый контроль		
В конце учебного года /или курса обучения/	<p>Определение изменения уровня развития детей, их творческих способностей.</p> <p>Определение результатов обучения.</p>	Защита проекта

4. КАЛЕНДАРНЫЙ ПЛАН ВОСПИТАТЕЛЬНЫХ МЕРОПРИЯТИЙ

Название мероприятия (тема)	Срок (месяц)
-----------------------------	--------------

Проведение классных часов и родительских собраний	Сентябрь 2023
Участие в конкурсах, олимпиадах	Февраль, март 2024
Защита проектов	Май 2024

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

В процессе обучения учащиеся не получают прямых оценок своей деятельности. Так как программа является развивающей, она не предполагает зачетно-экзаменационной системы контроля за результатами образования.

Контроль усвоения осуществляется педагогом на каждом занятии для коррекции своей педагогической деятельности. После решения задачи предполагается рефлексия, где каждый ребёнок высказывает своё мнение о том, что у него лучше всего получилось и над чем стоит поработать в дальнейшем. Во время проведения презентации проектов необходим подробный анализ положительных моментов и недочётов, при этом подчёркиваются позитивные стороны каждой ситуации.

Критерии результативности реализации программы

Уровень освоения программы	Критерии	Показатели
Высокий уровень	Учащиеся должны знать правила техники безопасности при работе, грамотно излагать программный материал, знать основные блоки команд, уметь самостоятельно применять изученные способы, аргументировать свою позицию, оценивать ситуацию и полученный результат.	Выполнение 100% - 70% заданий
Средний уровень	Учащиеся должны знать основные блоки команд, уметь самостоятельно или во взаимодействии с педагогом выполнять задания, уметь высказывать мнение, обобщать, классифицировать, обсуждать, грамотно и по существу излагать	Выполнение от 50% до 70% заданий

	программный материал, не допуская существенных неточностей в ответе.	
Низкий уровень	учащиеся не знают значительной части материала, допускают существенные ошибки, с большими затруднениями выполняют практические задания.	Выполнение менее 50% заданий

6. МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

В обучении школьников наиболее приемлемы комбинированные занятия, предусматривающие смену методов обучения и деятельности обучаемых, позволяющие свести работу за компьютером к регламентированной норме. С учетом данных о распределении усвоения информации и кризисах внимания учащихся на уроке, рекомендуется проводить объяснения в первой части занятия, а на конец – планировать деятельность, которая наиболее интересна для учащихся и имеет для них большее личностное значение. В комбинированном занятии можно выделить следующие основные этапы:

1. организационный момент;
2. активизация мышления и актуализация ранее изученного (разминка, короткие задания на развитие внимания, сообразительности, памяти, фронтальный опрос по ранее изученному материалу);
3. объяснение нового материала, сопровождаемое, как правило, компьютерной презентацией; на этом этапе учитель четко и доступно объясняет материал, по возможности используя традиционные и электронные наглядные пособия; учитель в процессе беседы вводит новые понятия, организует совместный поиск и анализ примеров, при необходимости переходящий в игру или в дискуссию;
4. работа за компьютером (выполнение работ компьютерного практикума, решение задач);
5. подведение итогов занятия.

На занятиях очень важно, чтобы каждый ученик имел доступ к компьютеру и пытался выполнять практические работы, пытался выполнять практические задания по описанию, обращаясь за помощью к учителю или товарищам.

Чтобы выполнение заданий компьютерного практикума шло успешно, школьников всякий раз следует подготовить к новому для них виду деятельности, подробно объяснив, что каждое задание выполняется в заданной последовательности и в строгом соответствии с описанием, поэтому нужно очень внимательно читать каждое указание

(каждый пункт), выполнять его, и только после этого переходить к следующему указанию (пункту). Нужно чтобы ученик очень четко осознавал, что он делает и какая именно операция у него не получается. Очень важно, чтобы учитель не подсказывал готовые решения, а, выявив истинную причину возникшего у ученика затруднения, направлял его к правильному решению.

Задания творческого характера представляют собой информационные мини-задачи. Выполнение творческого задания требует от ученика значительной самостоятельности при уточнении его условий, по представлению необходимой информации, по выбору технологических средств и приемов его выполнения

6. СПИСОК ЛИТЕРАТУРЫ

1. Поляков К.Ю. Программирование. Python. C++. Учебное пособие – М.: БИНОМ. Лаборатория знаний, 2019
- 2 Самоучитель Python. Выпуск 2.0. Дмитрий Мусин. – Май, 2017
- 3 Программирование для детей на языке Python — Москва: Издательство АСТ — 2017
4. Учим Python, делая крутые игры/ Эл Свейгарт [пер. с англ. М.А.Райтмана]. – Москва: Эксмо, 2018

□

ЭЛЕКТРОННЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ

1. <https://foxford.ru/> – Фоксфорд — онлайн-школа для учеников 1–11 классов, учителей и родителей.
2. <https://pythonworld.ru/> - Python для начинающих.
3. <https://younglinux.info/python.php> - Python. Введение в программирование.
- 4.