

**Муниципальное бюджетное общеобразовательное учреждение**

**«Средняя общеобразовательная школа № 9»**

Программа рассмотрена  
на заседании  
педагогического совета  
МБОУ СОШ №9

Протокол № 11  
от 13.05.2023 г.

Утверждено

Директор школы  
\_\_\_\_\_ Л.И. Шелковская

Приказ № 114/2 от  
13.05.2023 г.

**ДОПОЛНИТЕЛЬНАЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ  
ОБЩЕРАЗВИВАЮЩАЯ ПРОГРАММА**

**«Основы программирования на языке Python»**

Направленность: техническая  
Возраст обучающихся :13-14 лет

Составитель  
педагог дополнительного образования

Демкина Т.О.

г. Заполярный  
2023г.

## ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Python - современный язык программирования высокого уровня. Занимает лидирующие позиции в рейтинге языков программирования. Он простой по структуре, но очень гибкий

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» (далее – программа) -технической направленности.

Программа разработана в соответствии с нормативными документами:

- Закон РФ от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
- Приказ Министерства просвещения РФ от 27 июля 2022года №629 «Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам»;
- Распоряжение Правительства РФ от 31 марта 2022года N 678-р «Концепция развития дополнительного образования детей до 2030 года»;
- Приказ Министерства образования и науки Мурманской области от 19 марта 2020 года №462 «Об утверждении правил персонифицированного финансирования дополнительного образования детей Мурманской области»
- Постановление Главного государственного санитарного врача Российской Федерации от 28.09.2020 № 28, г. Москва «Об утверждении СанПиН 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодёжи». Санитарными правилами СП 1.2.3685-21 от 28.01.2021 № 2 «Гигиенические нормативы и требования к обеспечению безопасности и (или) безвредности для человека факторов среды обитания»;
- Уставом и локальными актами МБОУ СОШ №9.

Информация, информатика, информационные технологии — давно привычные для всех слова, которые очень точно характеризуют жизнь и потребности современного общества.

Информационные технологии (ИТ) – это совокупность методов и средств, используемых для сбора, хранения, обработки и распространения информации. В настоящее время деятельность человека стала сильно зависеть от этих технологии, они нуждаются в постоянном развитии.

В то же время можно утверждать о расслоении специалистов в сфере ИТ на тех, кто очень популярен и менее популярен среди работодателей. Это связано с тем, что некоторые отрасли в сфере информационных технологий пользуются особым приоритетом, и поэтому в них сосредоточено сравнительно больше ресурсов для развития.

Информационные технологии — быстро меняющаяся среда, в ней всегда много новшеств. Здесь практически ежедневно появляются самые разнообразные проекты и разработки.

Данная программа направлена, прежде всего, на удовлетворение индивидуальных образовательных интересов, потребностей и склонностей каждого школьника, на развитие творческих способностей, создание благоприятных условий для максимальной реализации творческого потенциала начинающих IT-специалистов. Программа поможет решить важную и сложную проблему осознанного профессионального выбора учащихся, расширение возможностей их социализации. Программа должна помочь учащимся узнать о разработках в области информатики и о множестве специалистов, которые трудятся в IT-сфере.

- специалистах, занимающихся компьютерным оборудованием и другими техническими разработками,
- специалистах, создающих программное обеспечение для компьютеров и других вычислительных устройств;
- специалистах, работающих с готовыми информационными продуктами.

**Актуальность программы** обусловлена тем, что в настоящее время одной из задач современного образования является содействие воспитанию нового поколения, отвечающего по своему уровню развития и образу жизни условиям информационного общества. Для этого обучающимся предлагается освоить основы программирования. А также для жизни в современном обществе необходимым является сформированное математическое мышление.

Размышления над задачами развивают интеллект, сообразительность, способствуют повышению уровня математической грамотности. Для обучения был выбран язык Python. Данный выбор обусловлен тем, что синтаксис языка достаточно прост и интуитивно понятен, а это понижает порог вхождения и позволяет сосредоточиться на логических и алгоритмических аспектах программирования, а не на выучивании тонкостей синтаксиса. При этом Python является очень востребованным языком; он отлично подходит для знакомства с различными современными парадигмами программирования и активно применяется в самых разных областях от разработки веб-приложений до машинного обучения.

Научившись программировать на языке Python, учащиеся получают мощный и удобный инструмент для решения как учебных, так и прикладных задач. Вместе с тем чистота и ясность его конструкций позволит учащимся потом с легкостью выучить любой другой язык программирования.

Занятие состоит из теоретической и практической частей. Во время первой преподаватель объясняет новый материал, приводит примеры его использования и рассказывает о его особенностях. После этого учащиеся задают появившиеся вопросы. Преподаватель приводит тестовый пример и объясняет его суть учащимся. Они в свою очередь наблюдают результаты выполнения программы на компьютере.

**Новизна** программы заключается в том, что существует необходимость постоянно обновлять и расширять профессиональные компетенции, также продиктована современными условиями информационного общества. Истинным профессионалам той отрасли науки и техники свойственно рассматривать умение представлять себя и свой продукт деятельности как инструмент, позволяющий

расширять и поддерживать профессиональную компетентность на должном уровне, улавливать самые перспективные тенденции развития мировой конъюнктуры, шагать в ногу со временем.

### **Педагогическая целесообразность образовательной программы**

Обучение по дополнительной общеобразовательной общеразвивающей программе «Основы программирования на языке Python» - это формирование новых общеинтеллектуальных умений и навыков: разделение задачи на этапы решения, построение алгоритма и др.

Как и другие языки программирования Python, служит для формирования мыслительных и психических процессов учащихся (внимание, память, логика), освоения приёмов умственных действий, самостоятельного нахождения и составления алгоритмов решения задач, умения строить модели, чётко и лаконично реализовывать этапы решения задач. Использование этих возможностей для формирования предметных и метапредметных результатов учащихся особенно важно, т.к. именно они активизирует процесс индивидуально-личностного становления учащихся. Программа поможет ребенку развить уровень логического и аналитического мышления, повысить успеваемость в школе по разным предметам.

Знания и умения, приобретенные в результате освоения программы, являются фундаментом для дальнейшего совершенствования мастерства в области программирования.

### **Уровень сложности освоения программы : «базовый»**

**Целью программы** является формирование у обучающихся устойчивых знаний в области программирования на языках, не входящих в школьный курс, но являющихся весьма востребованными в современном мире.

Программа направлена на развитие в ребенке устойчивого интереса к олимпиадному движению по информатике, освоению современных информационных технологий, проектной деятельности, практических навыков использования современной вычислительной техники, периферийных и мобильных устройств и других технических средств информатизации.

#### **Задачи программы**

Образовательные:

- знакомство обучающихся с углубленными знаниями в области теоретической информатики;
- приобщение ребенка к работе на компьютерном оборудовании для решения прикладных задач;
- изучение приемов программирования и решение практических задач.

Развивающие:

- формирование элементов самостоятельной интеллектуальной деятельности на основе овладения несложными математическими методами познания окружающего мира (умения устанавливать, описывать, моделировать и объяснять количественные и пространственные отношения);

- развитие основ логического, знаково-символического и алгоритмического мышления;
- развитие пространственного воображения;
- формирование системы математических знаний и умений их применять для решения учебно-познавательных и практических задач;
- развитие познавательных способностей;
- развитие умений аргументировано обосновывать и отстаивать высказанное суждение, оценивать и принимать суждения других;
- формирование у учащихся представлений о научном исследовании и опыта проектной деятельности.

Воспитательные:

- воспитание умения работать в мини-группе, культуры общения, ведения диалога и коммуникативных навыков;
- воспитание навыков здорового образа жизни;
- воспитание настойчивости, собранности, организованности, аккуратности;
- развить культуру труда, аккуратность и внимательность при обращении со сложной техникой.

### **Общая характеристика организации образовательного процесса.**

Условия реализации программы:

Возраст обучающихся, участвующих в реализации программы 13-14 лет.

Срок реализации программы :1 год

Объём программы – 36 часа.

Формы обучения – очная. Занятия – по группам. Состав группы – постоянный

Количество обучающихся в группе -15 человек

Режим занятий. Занятия проводятся – 1 раз в неделю по 1 академическому часу ( 45 минут).

**Планируемые результаты**, получаемые обучающимися в результате освоения программы.

Обучающиеся должны знать/понимать

- основные характерные черты информационного общества;
- основные компоненты информационной культуры человека;
- функции языка как способа представления информации;
- принципы кодирования информации,
- способы хранения и основные виды хранилищ информации,
- правила выполнения арифметических операций в двоичной системе счисления;
- основные логические операции; их свойства и обозначения,
- этапы информационной технологии решения задач с использованием компьютера;
- основные алгоритмические конструкции и уметь использовать для построения алгоритмов;

- операторы ввода, вывода, присваивания, условные и циклические операторы языка программирования,
- назначение и функции, используемых информационных и коммуникационных технологии;
- взаимосвязь учебного предмета с особенностями профессий и профессиональной деятельности, в основе которых лежат знания по данному учебному предмету,

Обучающиеся должны уметь

- приводить примеры получения, передачи и обработки информации в деятельности человека, живой природе, обществе и технике;
- представлять логические выражения в виде формул и таблиц истинности,
- строить логические схемы из основных логических элементов по формулам логических выражений;
- записывать на языке программирования алгоритм решения задачи и отлаживать
- работать с массивами: заполнять, осуществлять поиск и сортировку,
- осуществлять простейшую обработку цифровых изображений,
- искать информацию с применением правил поиска в компьютерных сетях, некомпьютерных источниках информации (справочниках, словарях, каталогах, библиотеках) при выполнении заданий и проектов по различным учебным дисциплинам;
- соблюдать правила техники безопасности, технической эксплуатации и сохранности информации при работе на компьютере;
- использовать приобретенные знания и умения в практической деятельности и повседневной жизни для создания информационных объектов, в том числе для оформления результатов учебной работы;
- организации индивидуального информационного пространства, создания личных коллекций информационных объектов;
- передачи информации по телекоммуникационным каналам в учебной и личной переписке, использования информационных ресурсов общества с соблюдением соответствующих правовых и эстетических норм.

#### **Формы диагностики результатов:**

- Тестирование;
- Форма итоговой аттестации – защита проекта.  
Итоговая аттестация может проводиться в форме отчетного выступления, защиты проекта или итогового тестирования. На основе примерных вопросов преподаватель может подготовить итоговый тест.  
Кроме этого в течение обучения воспитанники участвуют в олимпиадах и конференциях по профилю обучения, где происходит оценка их деятельности общественностью и специалистами.  
Критерии отбора работ (проектов):
- аккуратность исполнения;
- соблюдение технологии;
- творческий подход к работе.

Однако следует оговориться, что контроль знаний в группах должен осуществляться строго дифференцированно, исходя из возрастных,

физических, психологических особенностей развития каждого отдельного ребенка.

**Формы контроля:** наблюдение, устный опрос, тестирование, фронтальный опрос, самоконтроль, практическая работа, защита проекта, творческий отчёт. По завершении изучения крупных тем (в конце каждого модуля) планируется проведение нескольких занятий в форме презентации, где каждый ученик или группа учеников могут представить свои работы.

Способы контроля:

- устный опрос;
- комбинированный опрос;
- самопроверка;
- проверка в парах;
- проверка самостоятельной работы,
- игры;
- защита проектов;
- презентации.

По итогам обучения должно сформироваться представление о способе проведения научного исследования, актуальных задачах, самоопределение с областью дальнейшей проектно-исследовательской деятельности, а также должны быть сформированы следующие навыки: планировать и выполнять учебный проект, используя оборудование, модели, методы и приёмы, адекватные исследуемой проблеме. Уровень сформированной и освоенности навыков выявляется в ходе защит учебных проектных работ.

### УЧЕБНО-ТЕМАТИЧЕСКИЙ ПЛАН

	Содержание	Количество часов			Форма контроля
		Всего	Теория	Практика	
1	Знакомство с Python	1			
	Знакомство с сайтом <a href="http://tli1.net.io">tli1.net.io</a> Первый проект "Привет, Питон!" Второй проект "Случайное число" Третий проект "Десять случайных чисел»				
2	Отпечатываем черепашку на сцене	1			

	Отпечатываем черепашку на сцене Отпечатываем 10 черепашек в ряд Отпечатываем 10 черепашек в случайных местах сцены Отпечатываем 500 разноцветных черепашек в случайных местах сцены				Беседа по пройденной теме
3	Рисуем по координатам	1			

	Рисуем квадрат по координатам Рисуем треугольник по координатам Поднимаем и опускаем перо Рисуем пунктирную ЛШНИЮ				
4	Числа и текст в Python	1			
	Сложение чисел Сложение чисел и текста Преобразование текста в число Преобразование числа в текст				Практическое задание
5	Рисуем окружности	1			
	Рисуем гшцо Рисуем снеговика				Проведение опроса
6	Знакомство с модулем turtle	1			
	Рисуем квадрат Рисуем треугольник Рисуем букву М Рисуем ДОмик				Практическое задание
7	Знакомство с заливкой	1			
	Рисуем закрашенный круг Рисуем закрашенный треугольник Рисуем закрашенный квадрат Рисуем закрашенный кузов автомобиля				Проведение опроса
8	Рисуем с использованием переменных	1			Викторина



	Рисуем спираль Рисуем спираль из треугольников Рисуем окружность Рисуем згп-заг Рисуем закрашенный треугольник				
9	Рисуем с использованием функций	1			Проведение опроса
	Рисуем 30 квадратов с увеличивающейся стороной				Практическое задание

	Рисуем 30 симметричных квадратов с увеличивающейся стороной Рисуем дугу Рисуем цветок				
10	Пишем текст на сцене	1			
	ПГП_ПеМ ТУШ В июле ПГП_ПеМ ТУШ в случайных местах сцены Рисуем координатную ПЛОСКОСТЬ ,				Проведение опроса
11	Знакомство с RGB цветами	1			Практическое задание
	Выбор цвета RGB Рисуем гшнии снизу вверх Рисуем цветные линии снизу вверх. Изменяем цвет RGB.				
12	Знакомство с вещественными числами	1			Проведение опроса
	Арифметические операции с целыми и вещественными числами Порядок выполнения арифметических действий и сложные операции Сложекше строк Калькулятор				Практическое задание
13	Знакомство со срезами	1			

	<p>Знакомство со срезами          Выбираем букву из имени          Выбираем несколько букв из имени          Получение кода оператора из введенного номера телефона</p>				Проведение опроса
14	Методы для работы со строками	1			Практическое задание
	<p>Знакомство с методами          Вспоминаем срезы. Выделяем часы и минуты          Используем срезы и метод <code>find</code>.          Выделяем фамилию, имя и отчество</p>				

15	Вывод текста в рамке из символов	1			Проведение опроса
	<p>Символ переноса строки          Знакомство с методом <code>split</code>          Вывод ФИО в рамке</p>				Практическое задание
16	Знакомство с символом табуляции	1			
	<p>Символ табуляции <code>t</code>          Эксперименты с табуляцией          Рисуем ровные таблицы с помощью табуляции</p>				Проведение опроса
17	Знакомство с инструментами форматирования	1			Практическое задание
	<p>Инструменты форматирования. <code>%s</code>          вывод строки.          Инструменты форматирования. <code>%d</code>          вывод целых чисел.          Инструменты форматирования. <code>%f</code> вывод вещественных чисел.</p>		,	,	
18	Знакомство с кортежами	1			Проведение опроса

	Знакомство с кортежами Арифметические действия с кортежами Перебор элементов кортежа				Практическое задание
19	Знакомство со списками	1			
	Знакомство со списками Арифметические действия со списками Методы для работы со списками				Проведение опроса
20	ASCII Art	1			Практическое задание
	Знакомство с ASCII Ай Использование списка для создания ASCII Art Использование ЦИКЛа при создании ASCII Art				

21	Имена и профессии.	1			Проведение опроса
	Применение random.randint Выбор случайного элемента списка Составление фразы из нескольких случайных строк				Практическое задание
22	Строим фигуры по спискам координат	1			
	Вспоминаем списки Циклический перебор элементов списка Вспоминаем команды поднятия и опускания пера				Проведение опроса
23	Смайлики	1			Практическое задание
	Знакомство с кодами ASCII Знакомство с функцией ord() Знакомство с КАОМОДЗИ, повторяем срезы				

24	Построение фигур по координатам, хранящимся в одном списке	1			Проведение опроса
	Перебор списка в илюле с шагом Вспоминаем фунццо len() Вспоминаем команды работы с черепашкой				Практическ ое задание
25	Знакомство с методом joinO	1			
	Знакомство с методом join() Рисуем ASCII Art с помощью метода join().				Проведение опроса
26	Заполнение списка с координатами с клавиатуры и построение многоугольника.	1			Практическ ое задание
	Ввод данных с приглашением Использование метода append() Использование цикла при рисовании				
27	Рисование с помощью функций	1			Проведение опроса

	Знакомство с функцией posO Создание функций для рисования с помощью def Создание функций для рисования горизонтальньж и диагональньж линий из точки, в которой расположена черепашка				Практическ ое задание
28	Знакомство с объектами	1			
	Знакомство с объектами Параметры объектов Применение объектов				Проведение опроса
29	Программа с методом append, для ввода имен и профессий с клавиатуры.	1			

	Знакомство с методом append Заполнение густых списков пользователем Создание случайной фразы				Проведение опроса
30	Продвинутое использование методов split и rfind.	1			
	Применение метода splitO Применение метода rбnd() Вырезаем пмяфайла из строки				
31	Программируем Аквариум	1			
	Загрузка спрайтов Создание объектов сцена и черепашек Программирование спрайтов рыбок				Проведение опроса
32	Знакомство со словарями	1			
	Знакомство со словарями Работа со словарями				
	Типы данньж в словарю<				
33	Знакомство с юникодом	1			
	Знакомство с юникодом Создание рал,жи из стволов юникода Создание картины из СТоОЛОВ юникода				
34	Генераторы списков	1			Практическое задание
	Знакомство с генераторал,ш списков Применение генераторов списков Функция enumerate()				
35	Обобщающее повторение				
36	Обобщающее повторение				
	ВСЕГО	36	17	19	

## **Содержание программы**

### **Тема 1. Знакомство с Python**

- Знакомство с сайтом [trinket.io](https://trinket.io)
- Первый проект «Привет, Питон!»

- Второй проект «Случайное число»
- Третий проект «Десять случайных чисел»

### **Тема 2. Отпечатываем черепашку на сцене**

- Отпечатываем черепашку на сцене
- Отпечатываем 10 черепашек в ряд
- Отпечатываем 10 черепашек в случайных местах сцены
- Отпечатываем 500 разноцветных черепашек в случайных местах сцены

### **Тема 3. Рисуем по координатам**

- Рисуем квадрат по координатам
- Рисуем треугольник по координатам
- Поднимаем и опускаем перо
- Рисуем пунктирную линию

### **Тема 4. Числа и текст в Python**

- Сложение чисел
- Сложение чисел и текста
- Преобразование текста в число
- Преобразование числа в текст

### **Тема 5. Рисуем окружности**

- Рисуем лицо
- Рисуем снеговика

### **Тема 6. Знакомство с модулем turtle**

- Рисуем квадрат
- Рисуем треугольник
- Рисуем букву М
- Рисуем домик

### **Тема 7. Знакомство с заливкой**

- Рисуем закрашенный круг
- Рисуем закрашенный треугольник
- Рисуем закрашенный квадрат
- Рисуем закрашенный кузов автомобиля

### **Тема 8. Рисуем с использованием переменных**

- Рисуем спираль
- Рисуем спираль из треугольников
- Рисуем окружность
- Рисуем зигзаг
- Рисуем закрашенный треугольник

### **Тема 9. Рисуем с использованием функций**

- Рисуем 30 квадратов с увеличивающейся стороной
- Рисуем 30 симметричных квадратов с увеличивающейся стороной
- Рисуем дугу
- Рисуем цветок

### **Тема 10. Пишем текст на сцене**

- Пишем имя в цикле
- Пишем имя в случайных местах сцены
- Рисуем координатную плоскость.

### **Тема 11. Знакомство с RGB цветами**

- Выбор цвета RGB
- Рисуем линии снизу вверх
- Рисуем цветные линии снизу вверх. Изменяем цвет RGB.

### **Тема 12. Знакомство с вещественными числами**

- Арифметические операции с целыми и вещественными числами
- Порядок выполнения арифметических действий и сложные операции
- Сложение строк
- Калькулятор

### **Тема 13. Знакомство со срезами**

- Знакомство со срезами
- Выбираем букву из имени
- Выбираем несколько букв из имени
- Получение кода оператора из введенного номера телефона

### **Тема 14. Методы для работы со строками**

- Знакомство с методами
- Вспоминаем срезы. Выделяем часы и минуты
- Используем срезы и метод find. Выделяем фамилию, имя и отчество

### **Тема 15. Вывод текста в рамке из символов**

- символ переноса строки \n
- Знакомство с методом split
- Вывод ФИО в рамке

### **Тема 16. Знакомство с символом табуляции**

- символ табуляции t
- Эксперименты с табуляцией
- Рисуем ровные таблицы с помощью табуляции

### **Тема 17. Знакомство с инструментами форматирования**

- Инструменты форматирования. `0s` вывод строки.
- Инструменты форматирования. `0d` вывод целых чисел.
- Инструменты форматирования. `0f` вывод вещественных чисел.

### **Тема 18. Знакомство с кортежами**

- Знакомство с кортежами
- Арифметические действия с кортежами
- Перебор элементов кортежа

### **Тема 19. Знакомство со списками**

- Знакомство со списками
- Арифметические действия со списками
- Методы для работы со списками

### **Тема 20. ASCII Art**

- Знакомство с ASCII Art
- Использование списка для создания ASCII Art
- Использование цикла при создании ASCII Art

### **Тема 21. Имена и профессии**

- Применение `random.randint`
- Выбор случайного элемента списка



- Составление фразы из нескольких случайных строк

### **Тема 22. Строим фигуры по спискам координат**

- Вспоминаем списки
- Циклический перебор элементов списка
- Вспоминаем команды поднятия и опускания пера

### **Тема 23. Смайлики**

- Знакомство с кодами ASCII
- Знакомство с функцией ord()
- Знакомство с КАОМОДЗИ, повторяем срезы

### **Тема 24. Построение фигур по координатам, хранящимся в одном списке**

- Перебор списка в цикле с шагом
- Вспоминаем функцию len()
- Вспоминаем команды работы с черепашкой

### **Тема 25. Знакомство с методом pin(..)**

- Знакомство с методом join()
- Рисуем ASCII Art с помощью метода join().

### **Тема 26. Заполнение списка с координатами с клавиатуры и построение многоугольника**

- Ввод данных с приглашением
- Использование метода append()
- Использование цикла при рисовании

### **Тема 27. Рисование с помощью функций**

- Знакомство с функцией pos()
- Создание функций для рисования с помощью def
- Создание функций для рисования горизонтальных и диагональных линий из точки, в которой расположена черепашка

### **Тема 28. Знакомство с объектами**

- Знакомство с объектами
- Параметры объектов
- Применение объектов

### **Тема 29. Программа с методом append, для ввода имен и профессий с клавиатуры**

- Знакомство с методом append
- Заполнение пустых списков пользователем
  - Создание случайной фразы

### **Тема 30. Продвинутое использование методов split и rfind.**

- Применение метода split()
- Применение метода rfind()
- Вырезаем имя файла из строки

### **Тема 31. Програмируем Аквариум**

- Загрузка спрайтов
- Создание объектов сцена и черепашек
- Программирование спрайтов рыбок

### **Тема 32. Знакомство со словарями**

- Знакомство со словарями
- Работа со словарями

- Типы данных в словарж

### **Тема 33. Знакомство с юникодом**

- Знакомство с юникодом
- Создание рамки из символов юникода
- Создание картины из символов юникода

### **Тема 34. Генераторы списков**

- Знакомство с генераторами списков
- Применение генераторов списков
- Функция enumerate()

### **Тема 35-36 . Обобщающее повторение**

- Обобщающее повторение
- Обобщающее повторение

## **МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ**

В процессе реализации программы используются следующие педагогические технологии: технология личностно-ориентированного обучения, информационно-коммуникативная технология и здоровьесберегающая технология.

При личностно-ориентированном обучении на первое место выдвигается индивидуализация обучения - процесс раскрытия индивидуальности человека в специально организованной учебной деятельности. Ее цель состоит в том, чтобы учебно-познавательная деятельность учащихся обеспечивала их личностное самоопределение, развитие эмоционально-духовной сферы, формирование качеств. Данная технология применяется на протяжении всего учебного года при выполнении индивидуальных заданий.

ИКТ-технология раскрывается в том, что большинство заданий программы выполняется с помощью персонального компьютера и необходимых программных средств.

Под здоровьесберегающими образовательными технологиями в расширенном смысле можно понимать все те технологии, использование которых в образовательном процессе идет на пользу здоровья учащихся. Технология здоровье сбережения раскрывается через следующие методы и приемы: во время физкультминутки делаются физические упражнения для снятия общего напряжения и усталости глаз. Проводится регулярное проветривание кабинета. Наглядное пособие «Гимнастика для глаз», «Безопасность при работе за компьютером».

Типы занятий:

- комбинированный,
- первичное ознакомление с материалом;
- усвоение новых знаний;
- применение полученных знаний на практике; - закрепление, повторение;
- итоговое.

Формы организации учебного занятия: беседа, творческий отчет, урок-лекция, заочная экскурсия, деловая игра, практическое занятие, презентация.

Занятие проводится в форме тематического учебного занятия.

### **Дидактическое обеспечение**

- средства ИКТ (презентации, видеофильмы, обучающие игры, обучающие компьютерные программы, компьютеры, интерактивная доска, проектор);

- использование дидактического материала (карточки задания, схемы, таблицы, инструкции, практические задания);
- учебники, учебные пособия, журналы, книги;
- тематические подборки теоретического материала, игр, практических заданий;
- ресурсы сети Internet.

Использование собственного презентационного материала, собственных разработанных наглядных материалов. Занятия по программе предполагают наличие инструкции по технике безопасности и разработанные правила поведения в компьютерном классе для учащихся.

### **Материально-техническое обеспечение**

Занятия проходят в компьютерных классах, оснащенных необходимой мебелью, доской. Условия для занятий соответствуют санитарно-гигиеническим нормам. Кабинет оснащён компьютерами, проектором, интерактивной доской, что позволяет использовать для занятий видеофильмы, презентации, различные компьютерные программы.

Предполагается объединение компьютеров в локальную сеть с возможностью выхода в Интернет, что позволяет использовать сетевые цифровые образовательные ресурсы.

Каждое занятие делится на две части.

- дидактические игры и упражнения, разработка проекта, моделирование, планирование деятельности;
- работа в среде программирования Python непосредственно на компьютере. Для снятия утомления необходимо проводить физкультминутки.

Занятия проводятся в компьютерном классе. Предусматривается как индивидуальная работа учащихся, так и работа в группах.

### **Оценочные и методические материалы**

При обучении используются основные методы организации и осуществления учебно-познавательной работы, такие как словесные, наглядные, практические и проблемно-поисковые.

Выбор методов (способов) обучения зависит от психофизиологических, возрастных особенностей детей, темы и формы занятий. При этом в процессе обучения все методы реализуются в теснейшей взаимосвязи.

Данная программа предполагает следующие формы занятий Лекционно-практические занятия

Занятие, посвященное освоению новой теории, может начинаться как с мини-лекции педагога с введением новых понятий, так и с индивидуального решения детьми тематической проблемы.

В первом случае далее следует самостоятельное индивидуальное или коллективное решение детьми проблемы, подведение педагогом итогов с выявлением лучших путей решения.

Во втором случае после индивидуального решения проблемы детьми и индивидуального обсуждения решения с педагогом следует рассказ преподавателя о возможных способах выхода из ситуации, о возможных подходах к обсуждаемой теме и взаимосвязях с ранее изученными темами.

Задачи могут быть как теоретическими, так и экспериментальными. Возможна

комбинация этих подходов.

Лекция

Часть занятий проводится в форме лекции педагога, читающего теоретический материал.

Фронтальные или индивидуальные экспериментальные исследования.

Обсуждается новая теория и обучающимся предлагается ее проверить экспериментально. Возможны разнообразные комбинации предложенных методов.

Соревнование

Регулярно проводятся различные личные и командные соревнования: устные и письменные олимпиады. Данная форма работы в объединениях позволяет проводить оперативный мониторинг текущей успеваемости детей, вносит разнообразие в образовательной деятельности.

игра

Для развития навыков работы в группах, коллективного творчества, искусства ведения диспута проводятся различные физические игры и конкурсы. Наиболее распространенная форма игрового занятия тест, являющийся серьезным и насыщенным соревнованием.

Зачетные занятия

Изложение материала ведётся нетрадиционно, основным средством подачи материала являются презентации.

### **Материально-техническое обеспечение**

#### **Дидактическое обеспечение**

- Комплект учебно-методической документации;
- Тематические презентации;
- комплект электронных образовательных ресурсов;
- Презентации, обучающие компьютерные программы, компьютеры, проектор;
- Задания, схемы, таблицы, инструкции, практические задания и задачи;
- Ресурсы сети Интернет;
- Использование собственного презентационного материала, собственных разработанных наглядных материалов, иллюстраций.

#### **Материально-техническое обеспечение преподавателя:**

- Мебель: стол преподавателя, стул.
- Персональный компьютер или ноутбук с предустановленным программным обеспечением, камера, микрофон;
- Средства телекоммуникации - локальная сеть, выход в Интернет;
- Проектор мультимедийный;
- Экран;
- Презентации и видео по темам, дидактический материал;
- Методическое пособие

#### **Материально-техническое обеспечение ученика**

- Специализированная учебная мебель: компьютерные столы, стулья.
- Персональный компьютер с предустановленным программным обеспечением;

- Средства телекоммуникации - локальная сеть, выход в Интернет;
- Доступ к программному коду «Основы программирования на языке Python».

### **Критерии оценки:**

2 балла выставляется обучающемуся, если он полно излагает материал (отвечает на вопрос), дает правильное определение основных понятий; обнаруживает понимание материала, может обосновать свои суждения, применить знания на практике, привести необходимые примеры не только из учебника, но и самостоятельно составленные; излагает материал последовательно и правильно с точки зрения норм литературного языка;

1,5 балла выставляется обучающемуся, если он дает ответ, удовлетворяющий тем же требованиям, что и для выставления 2 баллов, но допускает 1–2 ошибки, которые сам же исправляет, и 1–2 недочета в последовательности и языковом оформлении излагаемого;

1 балл выставляется обучающемуся, если он обнаруживает знание и понимание основных положений данной темы, но излагает материал неполно и допускает неточности в определении понятий или формулировке правил; не умеет достаточно глубоко и доказательно обосновать свои суждения и привести свои примеры; излагает материал непоследовательно и допускает ошибки в языковом оформлении излагаемого;

0 баллов выставляется обучающемуся, если он обнаруживает незнание большей части соответствующего вопроса, допускает ошибки в формулировке определений и правил, искажающие их смысл, беспорядочно и неуверенно излагает материал.

### **ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО И ПРОМЕЖУТОЧНОГО КОНТРОЛЯ:**

Раздел (тема) дисциплины 1: «Введение в программирование на Python»

1. Что такое Python?
2. Какие бывают циклы в языке Python?
3. Для чего нужен механизм исключения?
4. Назовите четыре базовых типа данных в языке Python.
5. Почему они называются базовыми?
6. Что такое область видимости переменных?
7. Python — интерпретируемый язык или компилируемый?
8. Как использовать встроенные коллекции (list, set, dictionary)?
9. Что такое introspection?
10. Что такое индексы?

11. Что такое словарь в Python?
12. В чем разница между списком и кортежем?
13. Что такое функция map в Python?
14. Как создать пустой класс в Python?
15. Как преобразовать список в строку?

## Раздел (тема) дисциплины 2: «Основные стандартные модули Python»

1. Что понимается под понятием «модуль» в языке Python?
2. Какие бывают встроенные функции?
3. Назовите модули сервиса периода выполнения?
4. Перечислите функции для доступа к внутренним структурам?
5. Назовите числовые и строковые функции?
6. Какие бывают функции преобразования типов и классов?
7. Для чего нужен модуль os в языке Python?
8. Перечислите модули для хранения данных, архивации?
9. Что делает модуль random в языке Python?
10. Назовите модули для многопоточных вычислений?
11. Что такое PYTHONPATH?
12. Что такое pickling и unpickling?
13. Как бы вы произвели рандомизацию содержимого списка на месте?
14. Что такое pandas в Python?
15. Как получить текущий рабочий каталог с помощью Python?

## Раздел (тема) дисциплины 3: «Элементы функционального программирования»

1. Что такое функциональное программирование?
2. Что такое Рекурсия?
3. Расскажите про генераторы списков (list comprehension).
4. Как получить доступ к значениям в словаре?
5. Как добавить новое значение в объект списка?
6. Для чего нужны функции range() и xrange()?
7. Что такое итераторы в Python?
8. Как можно использовать тернарные операторы в Python?
9. Что такое функция map в Python?
10. Что такое абстракции у словаря и списка в Python?
11. Что означает ключевое слово self в Python?
12. Что такое операторы тождественности в Python?
13. В чем разница между тем, когда функция range() принимает один аргумент, два и три?
14. Как можно объявить несколько назначений в одной строке кода?
15. Как обрабатывать входные данные в Python?

## Раздел (тема) дисциплины 4: «Элементы ООП»

1. Перечислите принципы ООП?
2. Что такое абстракция?
3. Каковы основные принципы ООП (наследование, инкапсуляция, полиморфизм)?
4. Что такое декомпозиция?
5. Что понимается под атрибутом свойства (properties)?
6. Какие знаете принципы программирования?
7. Расскажите, какой в питоне механизм передачи параметров.
8. Какие есть типы объектов в питоне?
9. Что такое множественное наследование?
10. Что делает питон объектно-ориентированным?
11. Чем отличаются процедурная и объектно-ориентированная парадигмы программирования?
12. Что такое агрегация?
13. Что такое ассоциация?
14. Какие недостатки ООП?
15. Что такое приведение в Python?

## Раздел (тема) дисциплины 5: «Численные алгоритмы. Матричные вычисления»

1. Что такое массив, как его создать в Python?
2. Как добавить новое значение в объект списка?
3. Как объединить два массива NumPy?
4. В чем разница между списками (list) и кортежами (tuple)?
5. Являются ли массивы Python NumPy лучше списков?

6. Что такое отрицательный индекс (negative index)?
7. Как в питоне работает трёхместный (тернарный) оператор?
8. Что такое срез?
9. Как убрать из списка дубликат элемента?
10. Что вам известно про операторы сравнения в питоне?
11. Что такое словарь (dictionary)?
12. Сколько аргументов может принять range()?
13. Как работать с числами, которые не входят в десятичную систему счисления?
14. В чем заключается сложность доступа к элементам dict?
15. Освобождается ли вся память при выходе из Python?

## Раздел (тема) дисциплины 6: «Обработка текстов. Регулярные выражения. Unicode»

1. В каком случае одна и та же файловая переменная может быть использована для работы с несколькими файлами, а в каком – нет?
2. Назовите пример режимов обработки файлов с помощью Python?
3. Объясните «принцип сэндвича» при работе с файлами.
4. Как вы думаете, почему открытый программой файл, как правило, блокируется и другие программу не могут получить к нему доступ?
5. В чем разница между методами append () и extend ()?
6. В чем разница между дополнением и расширением?
7. Что такое «последовательный доступ к данным»?
8. Что желает модуль StringIO?
9. Для чего нужны регулярные выражения?
10. Как определить, что данные в файле закончились?
11. Расскажите, какой в питоне механизм передачи параметров.
12. Что такое «последовательный доступ к данным»?
13. Что такое файловая переменная? Почему для работы с файлом используют не имя файла, а файловую переменную?
14. Чем отличаются текстовые и двоичные файлы по внутреннему содержанию? Можно ли сказать, что текстовый файл – это частный случай двоичного файла?
15. В каких случаях нужно открывать одновременно несколько файлов?

## Раздел (тема) дисциплины 7: «Работа с данными в различных форматах»

1. Что такое модуль Numeric?
2. Как проверить, что строка содержит только цифры?
3. Как лучше всего поменять местами числовые значения объектов?
4. Какие есть типы данных и в чем разница между list и tuple, зачем они?
5. В чем разница между списками и кортежами с точки зрения синтаксиса?
6. Что такое пространство имен в Python?
7. Что такое лямбда в спецификации Python?



8. Как получить данные от пользователя в Python?
9. Какой метод используется для определения местоположения указателя в файле?
10. Как передаются аргументы функций в Python (by value or reference)?
11. Что значит \*args, \*\*kwargs? И зачем нам их использовать?
12. Что такое нарезка в Python?
13. Как проверить, что строка содержит только цифры и буквы
14. Как можно удалить все пробелы из строки?
15. Что делает модуль RandomArray?

## Раздел (тема) дисциплины 8: «Многопоточные вычисления»

1. Что такое многопоточность?
2. Что такое битовые операторы?
3. Что такое цикломатическая сложность?
4. Назовите функции модуля threading
5. Какие модули необходимы для многопоточного программирования?
6. Напишите программу на Python, чтобы создать треугольник из звезд
7. Как работает thread locals?
8. Является ли функция допустимой, если она не имеет оператора return?
9. Какими методами можно управлять жизнью потоков?
10. Подключите конструкции Looping, доступные в Python.
11. Что такое Семафоры?
12. В чем разница между методом input () и методом raw\_input ()?
13. Для чего нужен таймер?
14. Когда нужны замки?
15. Приведите пример многопоточной программы

## Раздел (тема) дисциплины 9: «Создание приложений с GUI»

1. Что такое Tkinter в Python и для чего он используется?
2. Как создать элемент метки в Python?
3. Что такое фрейм в графическом интерфейсе Python?
4. Расскажите про некоторые элементы графического интерфейса в модуле Python Tkinter?
5. Как создать элемент кнопки проверки в Python?

6. Какой метод используется для установки указателя файла в определенное место?
7. В чем разница между элементом Text Entry и Text Box в модуле Tkinter?
8. Как вы определяете размеры окна в графической программе Python?
9. Для чего в Python используется метод grid ()?
10. Как создать элемент текстового поля в Python?
11. Как создать элемент кнопки в Python?
12. Что такое PEP 8 в Python?
13. Как мы можем импортировать разные пакеты в программу Python?
14. Обязательны ли отступы в Python?

## СПИСОК ЛИТЕРАТУРЫ

### Список литературы и ресурсы сети Интернет для педагога

1. <https://shultais.education/courses/python-3> - Online-курс программирования на языке Python
2. <https://pythonworld.ru/samouchitel-python> - Самоучитель Python 3, собранный из материалов данного сайта. Предназначен в основном для тех, кто хочет изучить язык программирования Python с нуля.
3. Python 3 для начинающих [Электронный ресурс]. — Режим доступа. pythonworld.ru.
4. Питонтьютор [Электронный ресурс]. — Режим доступа: pythontutor.ru.

Литература для ученика:

1. Марк Лутц «Python. Карманный справочник», Издательство: Вильямс, 2016
2. Марк Лутц «Изучаем Python», Издательство: Символ-плюс, 2011
3. МакГрат Майк «Программирование на Python для начинающих», Издательство: Эксмо, 2015

### Календарный учебный график

Год обучения	Дата начала обучения по программе	Дата окончания обучения по программе	Всего учебных недель	Кол-во учебных часов по программе	Режим занятий
1 год	01.09..2023 В соответствии с расписанием	20.05.2024 В соответствии с расписанием	34	36	1 раз по 1 академическому часу (45 минут)

	занятий	занятий			
--	---------	---------	--	--	--