

**МУНИЦИПАЛЬНОЕ ОБЩЕОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
«СРЕДНЯЯ ОБЩЕОБРАЗОВАТЕЛЬНАЯ ШКОЛА № 10» г. УХТЫ**

Согласовано

Педагогическим советом
МОУ «СОШ № 10» г. Ухты

Протокол от 20.03.2023 № 1

Утверждено

приказом директора
МОУ «СОШ № 10» г. Ухты
от 21.03.2023 № 01-13/ 79

Дополнительная общеобразовательная общеразвивающая программа
«Основы программирования на языке Python»
(базовый уровень)

Направленность: техническая

Адресат программы: учащиеся от 13 до 14 лет

Срок реализации: 2 года

Автор-составитель:

Озерова Е.А., учитель
информатики

Ухта

2023

Оглавление

Раздел №1 Комплекс основных характеристик дополнительной общеобразовательной общеразвивающей программы	3
Пояснительная записка.....	3
Задачи программы:	6
Учебный план программы «Основы программирования на языке Python».....	8
Содержание программы	12
Первый год обучения (68 часов)	12
Второй год обучения (68 часов)	13
Планируемые результаты.....	15
 Раздел № 2. Комплекс организационно-педагогических условий.....	17
Условия реализации программы	17
Формы итоговой аттестации – защита проекта.....	19
Методические материалы	20
Список литературы	22
Электронные образовательные ресурсы.....	23

Раздел №1 Комплекс основных характеристик дополнительной общеобразовательной
общеразвивающей программы

Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» (техническое направление) разработана на основе:

1. Федерального закона от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации»;
2. Концепции развития дополнительного образования детей, утвержденной Распоряжением Правительства Российской Федерации от 4.09.2014 № 1726-р;
3. СанПин 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей», Утвержденных постановлением Главного государственного санитарного врача Российской Федерации от 4.07.2014 № 41;
4. Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», утвержденным приказом Министерства просвещения России от 09.11.2018 г. № 196 г.;
5. Методическими рекомендациями по проектированию дополнительных общеобразовательных программ (включая разноуровневые программы) (Письмо Министерства образования и науки России от 18.11.2015 №09-3242);
6. Методических рекомендаций по проектированию дополнительных общеобразовательных – дополнительных общеразвивающих программ в Республике Коми (Письмо Министерства образования и молодежной политики Республики Коми от 27.01.2016 №07-27/45)
7. Правилами персонифицированного финансирования дополнительного образования детей, утвержденными приказом Министерством образования, науки и молодежной политики Республики Коми от 01.06.2018 №214-п;
8. Рекомендациями «Проектирование дополнительных общеобразовательных общеразвивающих программ» (включая разноуровневые и модульные программы) ГАУДО «РЦДО».

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» имеет *техническую направленность*.

Программа «Основы программирования на языке Python» учитывает возрастные и индивидуальные особенности учащихся и направлена на:

- создание необходимых условий для личностного развития учащихся, позитивной социализации и профессионального самоопределения;
- удовлетворение индивидуальных потребностей, учащихся в интеллектуальном,

техническом развитии;

- формирование и развитие творческих способностей учащихся, выявление, развитие и поддержку талантливых учащихся.

Новизна программы заключается в том, что её реализация будет осуществляться с помощью оборудования школьного Кванториума, обеспечивающего развитие у обучающихся современных компетенций и навыков, в том числе технической, математической, информационной грамотности, формирования критического и креативного мышления, а также повышения качества образования.

Актуальность программы «Основы программирования на языке Python» вызвана потребностью современного информационного общества в высокообразованных, адаптированных к изменениям специалистах в IT-сфере. Для удовлетворения данной потребности перед дополнительным образованием стоит задача развития человеческого потенциала через выявление талантливых детей, развитие их мотивации и способностей.

Изучение языка программирования Python поможет ребенку получить более целостное представление о профессии программиста, разработчика, инженера.

Программа предполагает участие обучающихся в интеллектуальных соревнованиях по программированию различных уровней, создание образовательных практических или научно-исследовательских проектов, что даст возможность детям полностью реализовать свой интеллектуальный и творческий потенциал.

Педагогическая целесообразность. Данная программа способствует развитию 4к — компетенций детей (коммуникация, креативность, командная работа, критическое мышление.), тем самым отвечая потребностям общества и федеральному государственному образовательному стандарту. В программе реализуются системный, комплексный, личностно-ориентированный и теоретический подходы к развитию детей. Адаптация материала соответствует возрастным и психофизиологическим особенностям детей.

В процессе алгоритмизации и программирования задач, обучающиеся получают дополнительные знания в области математики и информатики, что, в конечном итоге, изменит картину восприятия обучающимися технических дисциплин, переводя их из разряда умозрительных в разряд прикладных.

Основные принципы алгоритмизации и программирования послужат хорошей почвой для подготовки к экзаменам и последующего освоения более сложного теоретического материала на занятиях.

Возможность самостоятельной разработки программного продукта для обучающихся в современном мире является очень мощным стимулом к познанию нового и формированию стремления к самостоятельному созиданию, способствует развитию уверенности в своих силах и расширению горизонтов познания. Занятия по программе «Основы программирования на языке Python» позволяют заложить фундамент для подготовки будущих специалистов нового склада, способных к совершению инновационного прорыва в современной науке и технике.

Отличительная особенность данной программы в том, что носит практико-ориентированный характер. Программа личностно-ориентирована и составлена так, чтобы каждый ребёнок имел возможность самостоятельно выбрать наиболее интересный объект работы, приемлемый для него.

Адресат программы. Программа «Основы программирования на языке Python» ориентирована на детей подросткового и старшего школьного возраста: 13 – 14 лет.

Условия набора учащихся:

Прием учащихся осуществляется на основе письменного заявления родителей (законных представителей). На обучение принимаются все желающие учащиеся, не имеющие медицинских противопоказаний.

Уровень освоения программы: базовый.

Объем программы: 136 часов.

Срок освоения программы: программа рассчитана на 2 года обучения

Форма обучения – очная.

Режим занятий: продолжительность занятия - 40 минут, 2 часа в неделю.

Форма организации: групповые занятия, количество учащихся в группах: 8 - 12 человек. Групповая форма занятий позволяет педагогу построить процесс обучения в соответствии с принципами дифференцированного и индивидуального подходов.

Занятия по программе состоят из теоретической и практической частей, причем большее количество времени занимает практическая часть.

Цель программы – создание условий для формирования у учащихся комплекса компетенций в области алгоритмизации и программирования на языке «Python» (навыков алгоритмического и логического стиля мышления, представления о приемах и методах программирования через составление алгоритмов и обучение искусству программирования), а так же с ориентацией на получение специальностей, связанных с программированием, через формирование ключевых компетенций, основанных на создании ценностно ориентированного, конструктивного стиля мышления и новых способах самостоятельной творческой деятельности по направлению IT.

Задачи программы:

Обучающие:

- способствовать развитию базовой части математического аппарата, применяемого в современном программировании;
- обучить навыкам работы в интегрированных средах разработки и в онлайн сервисах Интернета, связанных с программированием;
- развить культуру и обучить навыкам алгоритмического и логического мышления, грамотной разработки программ;
- обучить школьников навыкам разработки эффективных алгоритмов и программ на основе языка программирования Python, как методу, предполагающему создание понятных программ, обладающих свойствами модульности;
- способствовать возможностям получения новых знаний в области компьютерного программирования;
- формировать у учащихся интерес к профессиям, требующим навыков алгоритмизации и программирования;

Развивающие:

- развивать навыки проектного мышления, работы в команде, эффективно распределять обязанности при реализации проекта, требующего от участников знаний и умений из различных направлений;
- способствовать развитию эстетического вкуса, конструктивного и критического мышления;
- развивать навыки инженерного мышления и умения работать по предложенным инструкциям;
- развивать навыки программирования, проектирования и эффективного использования электронного вычислительного оборудования;
- развивать внимательность, аккуратность и изобретательность при выполнении учебных проектов;
- способствовать развитию внимания, творческого, логического и абстрактного мышления, воображения, умение излагать мысли в четкой логической последовательности, отстаивать свою точку зрения, анализировать ситуацию и самостоятельно находить ответы на вопросы путем логических рассуждений и информационного поиска.
- способствовать развитию творческих авторских начал через создание самостоятельных программ, участие в конкурсах и олимпиадах;

- способствовать обогащению жизненного опыта и формированию системы ценностей.

Воспитательные:

- способствовать формированию правильного методологического подхода к познавательной и практической деятельности;
- способствовать формированию мотивации учащихся к изобретательству, созданию собственных программных реализаций;
- способствовать стремлению к овладению техникой исследования;
- воспитывать трудолюбие, инициативность и настойчивость в преодолении трудностей;
- создавать условия для стремления к получению качественного законченного результата в проектной деятельности;
- развивать навыки сознательного и рационального использования компьютера в своей учебной, а впоследствии и в профессиональной деятельности;
- развивать раннюю ориентацию на инновационные технологии и методы организации практической деятельности;
- воспитывать взаимоуважение друг к другу, эстетический вкус, бережное отношение к оборудованию и технике, дисциплинированность.

При проведении занятий используются следующие формы проведения занятий:

- практическое занятие - основная форма проведения занятий, обучающиеся выполняют задания в течении части задания или нескольких занятий;
- фронтальная (получение обучающимися нового материала);
- индивидуальная;
- групповая (работа в малых группах);
- презентация (презентация новых техник и технологий, новых проектов);
- проектная деятельность (получение новых знаний, реализация личностных проектов, защита проектов);
- олимпиада (выполнение олимпиадных заданий Всероссийских дистанционных олимпиад);
- открытое занятие;
- соревнование;
- викторина;
- защита проектов.

**Учебный план программы «Основы программирования на языке Python»
Первый год**

№ п/п	Название раздела, темы	Кол-во часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
		68	13	55	
Раздел 1. Линейные и разветвляющиеся алгоритмы.					
1	История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы.	1	1		
2	Вывод, типы данных и переменных. Определение переменной. Переменные и арифметические выражения.	2	1	1	
3	Чтение данных. Операции над строками. Примеры решения задач.	3	1	2	
4	Отработка навыков решения простейших задач.	4		4	Практические задания
5	Логический тип данных и операции. Примеры использования логических выражений.	1	1		
6	Арифметика строк. Арифметика чисел	3	1	2	
7	Условный оператор. Вложенный условный оператор. Операции сравнения.	5	1	4	
8	Отработка навыков решения простейших задач (условный оператор, операции сравнения)	6		6	Практические задания
Раздел 2. Циклические алгоритмы					
1	Цикл WHILE. Примеры решения задач.	4	1	3	
2	Цикл с параметром. Переменная цикла for. Варианты цикла for	5	1	4	

3	Подсчет суммы и оператор CONTINUE. Примеры решения задач.	2	1	1	
4	Отработка навыков решения простейших задач.	3		3	Практические задания
5	Вещественные числа. Основы работы с вещественными числами. Округление вещественных чисел. Примеры решения задач.	3	1	2	
6	Отработка навыков решения простейших задач.	3		3	Практические задания
Раздел 3. Строки. Списки					
1	Срезы строк. Использование срезов. Индексы строк. Метод FIND. Примеры решения задач.	4	1	3	
2	Отработка навыков решения простейших задач.	3		3	Практические задания
3	Методы RFIN, REPLACE и COUNT. Примеры решения задач.	2	1	1	
4	Отработка навыков решения простейших задач.	3		3	Практические задания
5	Списки. Метод SPLIT и JOIN. Примеры решения задач.	3	1	2	
6	Отработка навыков решения задач.	6		6	Практические задания
7	Заключительное занятие	2		2	Практические задания

Второй год

№ п/п	Название раздела, темы	Кол-во часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
		68	14	54	
Повторение: линейные, разветвляющиеся, циклические алгоритмы, строки, списки.					
1	Повторение. Основные конструкции языка программирования Python. Типы данных. Арифметические выражения. Условный оператор. Циклы. Решение задач.	4	1	2	
2	Срезы строк. Методы. Решение задач.	3	1	2	
3	Списки. Методы работы со списками. Обработка списка. Примеры решения задач.	2	1	1	
4	Отработка навыков решения задач.	4		4	Практические задания
5	Решение несложных олимпиадных задач.	4		4	Практические задания
Кортежи					
1	Кортежи. Функция RANGE, цикл FOR. Примеры решения задач.	2	1	1	
2	Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. Примеры решения задач.	2	1	1	
3	Отработка навыков решения задач.	4		4	Практические задания
Функции. Множества. Словари.					
1	Функции. Использование функций. Примеры решения задач.	2	1	1	
2	Рекурсия. Использование рекурсии. Примеры решения задач.	2	1	1	

3	Возврат значений. Локальные и глобальные переменные. Примеры решения задач.	2	1	1	
4	Структуры в Python. Лямбдафункции. Именованные параметры.	2	1	1	
5	Сортировка подсчетом. Примеры решения задач.	2	1	1	
6	Отработка навыков решения задач.	4		4	Практические задания
7	Множества и хеш-функции. Создание и работа с множествами.	2	1	1	
8	Отработка навыков решения задач.	2		2	Практические задания
9	Решение олимпиадных задач.	3		3	Практические задания
10	Словари. Методы строк. Примеры решения сложных задач.	4	2	2	
Файлы					
1	Работа с файлами. Чтение из файла. Запись в файл.	1	1		
2	Менеджеры контекста with .. as	1		1	
3	Документирование кода в Python	1		1	
4	Работа с модулями: создание и подключение инструкций import .. from	1		1	
5	Отработка навыков решения сложных задач.	6		6	Практические задания
6	Решение сложных олимпиадных задач.	6		6	Практические задания
7	Заключительное занятие	2		2	Практические задания

Содержание программы

Первый год обучения (68 часов)

Вводное занятие. Правила техники безопасности

Раздел 1. Линейные и разветвляющиеся алгоритмы.

Тема 1.1. История языков программирования. Введение в язык программирования Python. Среда программирования Python. Установка программы. Теория (1 ч.)

Тема 1.2. Вывод, типы данных и переменных. Определение переменной. Переменные и арифметические выражения. Теория (1 ч.). Практика (1 ч.)

Тема 1.3. Чтение данных. Операции над строками. Примеры решения задач. Теория (1 ч.). Практика (2 ч.)

Тема 1.4. Отработка навыков решения простейших задач. Практика (4 ч.) Форма подведения итогов: практические задания.

Тема 1.5. Логический тип данных и операции. Примеры использования логических выражений. Теория (1 ч.).

Тема 1.6. Арифметика строк. Арифметика чисел. Теория (1 ч.). Практика (1 ч.)

Тема 1.7 Условный оператор. Вложенный условный оператор. Примеры решения задач. Теория (1 ч.). Практика (4 ч.)

Тема 1.8. Отработка навыков решения простейших задач. Практика (6 ч.) Форма подведения итогов: практические задания.

Раздел 2. Циклические алгоритмы.

Тема 2.1. Цикл WHILE. Примеры решения задач. Теория (1 ч.). Практика (3 ч.)

Тема 2.2. Цикл с параметром. Переменная цикла for. Варианты цикла for. Теория (1 ч.). Практика (4 ч.)

Тема 2.3. Подсчет суммы и оператор CONTINUE. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 2.4. Отработка навыков решения простейших задач. Практика (3 ч.) Форма подведения итогов: практические задания.

Тема 2.5. Вещественные числа. Основы работы с вещественными числами. Округление вещественных чисел. Примеры решения задач. Теория (1 ч.). Практика (2 ч.) Тренировочные задания.

Тема 2.6. Отработка навыков решения простейших задач. Практика (3 ч.) Форма подведения итогов: практические задания.

Раздел 3. Строки. Списки.

Тема 3.1. Срезы строк. Использование срезов. Метод FIND. Примеры решения задач. Теория (1 ч.). Практика (3 ч.)

Тема 3.2. Отработка навыков решения простейших задач. Практика (3 ч.) Форма подведения итогов: практические задания.

Тема 3.3. Методы RFIND, REPLACE и COUNT. Примеры решения задач. Теория(1 ч.). Практика (1 ч.)

Тема 3.4. Отработка навыков решения простейших задач. Практика (3 ч.)
Форма подведения итогов: практические задания.

Тема 3.5. Списки. Метод SPLIT и JOIN. Примеры решения задач. Теория (1 ч.). Практика (2 ч.)

Тема 3.6. Отработка навыков решения задач. Практика (6 ч.) Форма подведения итогов: практические задания.

Тема 3.7. Заключительное занятие. Практика (2 ч.) Форма подведения итогов: практические задания.

Второй год обучения (68 часов)

Раздел 1. Повторение: линейные, разветвляющиеся, циклические алгоритмы, строки, списки.

Тема 1.1. Повторение. Основные конструкции языка программирования Python. Типы данных. Арифметические выражения. Условный оператор. Циклы. Решение задач. Теория (1 ч.). Практика (3 ч.)

Тема 1.2. Срезы строк. Методы. Решение задач. Теория (1 ч.). Практика (2 ч.)

Тема 1.3. Списки. Методы работы со списками. Обработка списка. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 1.4. Отработка навыков решения задач. Практика (4 ч.) Форма подведения итогов: практические задания.

Тема 1.5. Решение несложных олимпиадных задач. Практика (4 ч.) Форма подведения итогов: практические задания.

Раздел 2. Кортежи.

Тема 2.1. Кортежи. Функция RANGE, цикл FOR. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 2.2. Сортировка. Сравнение списков и кортежей. Именованный параметр KEY. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 3.6. Отработка навыков решения задач. Практика (4 ч.) Форма подведения итогов: практические задания.

Раздел 3. Функции. Множества. Словари.

Тема 3.1. Функции. Использование функций. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 3.2. Рекурсия. Использование рекурсии. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 3.3. Возврат значений. Локальные и глобальные переменные. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 3.4. Структуры в Python. Лямбда-функции. Именованные параметры. Теория (1 ч.). Практика (1 ч.)

Тема 3.5. Сортировка подсчетом. Примеры решения задач. Теория (1 ч.). Практика (1 ч.)

Тема 3.6. Отработка навыков решения задач. Практика (4 ч.) Форма подведения итогов: практические задания.

Тема 3.7. Множества и хеш-функции. Создание и работа с множествами. Теория (1 ч.). Практика (1 ч.)

Тема 3.8. Отработка навыков решения задач. Практика (2 ч.) Форма подведения итогов: практические задания.

Тема 3.9. Решение олимпиадных задач. Практика (3 ч.) Форма подведения итогов: практические задания.

Тема 3.10. Словари. Методы строк. Примеры решения сложных задач. Теория (2 ч.). Практика (2 ч.)

Раздел 4. Файлы.

Тема 4.1. Работа с файлами. Чтение из файла. Запись в файл. Теория (1 ч.)

Тема 4.2. Менеджеры контекста with .. as. Практика (1 ч.)

Тема 4.3. Документирование кода в Python. Практика (1 ч.)

Тема 4.4. Работа с модулями: создание и подключение инструкций import .. from. Практика (1 ч.)

Тема 4.5. Отработка навыков решения сложных задач. Практика (6 ч.) Форма подведения итогов: практические задания.

Тема 4.6. Решение сложных олимпиадных задач. Практика (6 ч.) Форма подведения итогов: практические задания.

Тема 4.7. Заключительное занятие. Практика (2 ч.) Форма подведения итогов: практические задания.

Планируемые результаты

В ходе реализации программы «Основы программирования на языке Python» должны быть созданы условия для достижения следующих результатов:

Личностные результаты:

Обучающиеся будут демонстрировать в деятельности:

- критическое отношение к информации и избирательность её восприятия;
- готовность и способность к саморазвитию и личностному самоопределению;
- умение организовывать свою деятельность (планирование, контроль, оценка);
- способность к самостоятельным действиям, ответственность за их результаты;
- готовность открыто выражать и отстаивать свою позицию;
- коммуникативную компетентность в общении и сотрудничестве со сверстниками;
- понимание основ экологической культуры, соответствующей современному уровню экологического мышления.

Метапредметные результаты:

Обучающиеся будут демонстрировать в деятельности:

- овладение составляющими исследовательской и проектной деятельности, включая умения видеть проблему, ставить вопросы, выдвигать гипотезы, давать определения понятиям, наблюдать, проводить эксперименты, делать выводы и заключения, структурировать материал, объяснять, доказывать, защищать свои идеи;
- готовность оценивать получающийся творческий продукт и соотносить его с изначальным замыслом, выполнять по необходимости коррекции либо продукта, либо замысла
- способность самостоятельно определять понятия, создавать обобщения, устанавливать аналогии, классифицировать, самостоятельно выбирать основания и критерии для классификации, устанавливать причинно-следственные связи, строить логическое рассуждение, умозаключение (индуктивное, дедуктивное и по аналогии) и делать выводы, самостоятельно формулировать вопросы проблемного и исследовательского характера;
- способность организовывать учебное сотрудничество и совместную деятельность с педагогом и сверстниками, эффективную индивидуальную и групповую работу, аргументацию и защиту своего мнения, грамотное использование коммуникационно-информационных средств для достижения поставленной цели и разрешение

конфликтов на основе согласования позиций и учета интересов.

Предметные результаты

Обучающиеся будут демонстрировать в деятельности:

- составлять алгоритмы для решения учебных задач различных типов;
- выражать алгоритм решения задачи различными способами (словесным, графическим, в том числе и в виде блок-схемы, с помощью формальных языков и др.);
- определять наиболее оптимальный способ выражения алгоритма для решения конкретных задач (словесный, графический, с помощью формальных языков);
- определять результат выполнения заданного алгоритма (программы) или его фрагмента;
- использовать термины «исполнитель», «алгоритм», «программа», а также понимать разницу между употреблением этих терминов в обыденной речи и в информатике;
- выполнять без использования компьютера («вручную») несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных, записанные на конкретном языке программирования с использованием основных управляющих конструкций последовательного программирования (линейная программа, ветвление, повторение, вспомогательные алгоритмы);
- составлять несложные алгоритмы управления исполнителями и анализа числовых и текстовых данных с использованием основных управляющих конструкций последовательного программирования и записывать их в виде программ на выбранном языке программирования; выполнять эти программы на компьютере;
- использовать величины (переменные) различных типов, табличные величины (массивы), а также выражения, составленные из этих величин; использовать оператор присваивания;
- анализировать предложенный алгоритм, например, определять какие результаты возможны при заданном множестве исходных значений;
- использовать логические значения, операции и выражения с ними;
- записывать на выбранном языке программирования арифметические и логические выражения и вычислять их значения.

Раздел № 2. Комплекс организационно-педагогических условий

Условия реализации программы

Материально-техническое обеспечение:

Для проведения лекций и мини-конференции предусмотрен кабинет, оснащенный компьютерной техникой, не менее 1 ПК на двух обучающихся, проектором, экраном, магнитно-маркерной доской, магнитно-маркерным флип-чартом.

Практические занятия курса «Основы программирования на языке Python» проводятся в учебном кабинете, предназначенном для подготовки и проведения занятий по программированию. Оборудование и техника работ в учебной лаборатории должны соответствовать требованиям, предъявляемым к производственным и другим лабораториям соответствующего профиля.

Учебно-методические средства обучения: применяемое на занятиях дидактическое и учебно-методическое обеспечение включает в себя электронные учебники, справочные материалы и системы используемых Программ, Интернет, рабочие тетради обучающихся.

Специальное оборудование школьного Кванториума:

- Ноутбуки по количеству учащихся;
- Многофункциональное устройство (МФУ);
- Флипчарт.

Информационно-методическое обеспечение

1. Мультимедийные презентации по всем модулям и темам для сопровождения занятий;
2. Разработанные конспекты лекционных занятий;
3. Иллюстративный материал по всем темам;
4. Методические указания по организации практических работ;
5. Тематика проектных работ;
6. Информационная и справочная литература.

Кабинет, в котором проводятся занятия, должен быть оборудован удобной мебелью, шкафами и стеллажами для хранения пособий и учебных материалов, наглядными пособиями, доской.

Каждый учащийся обеспечивается доступом к библиотечным фондам и фондам аудио и видеозаписей школьной библиотеки. Во время самостоятельной работы учащиеся могут пользоваться Интернетом для сбора дополнительного материала.

Кадровое обеспечение

Дополнительная общеобразовательная общеразвивающая программа «Основы программирования на языке Python» реализуется лицами, имеющими высшее образование по направлению «техническое». Для успешной реализации данной программы необходимы педагоги, способные к инновационной профессиональной деятельности, обладающие необходимым уровнем методологической культуры и сформированной готовностью к непрерывному образованию в течение всей жизни.

Формы итоговой аттестации – защита проекта.

Оценочные материалы. Диагностика результативности образовательного процесса.

Система оценки и фиксирования результатов

В начале учебного года при комплектовании групп осуществляется входной контроль (в форме собеседования и стартового тестирования) для определения уровня развития детей и их творческих способностей и распределения по группам базового и продвинутого уровня.

Аттестацию учащихся в процессе реализации программ рекомендуется проводить с использованием диагностических методов.

Цель проведения диагностики – определение изменения уровня развития учащихся, их творческих способностей, получение сведений для совершенствования образовательной программы и методов обучения.

Промежуточная диагностика или текущий контроль позволяет выявить и проанализировать уровень усвоения материала реализуемого модуля и внести необходимые коррективы, в том числе и индивидуально.

Итоговая диагностика является необходимым завершающим элементом программы и проводится при завершении реализации программы. В качестве итоговой формы аттестации рекомендуется проведение научно-практической конференции, на которой учащиеся представляют результаты проектной деятельности.

Одним из показателей результативности реализации программ технической направленности является участие в выставках, конкурсах, конференциях муниципального, регионального, федерального уровней. Формой фиксации результатов в данном случае являются свидетельства (сертификаты участия), грамоты и дипломы, портфолио, статьи в информационных источниках).

Сравнительный анализ различных форм контроля позволяет оценить в целом достижение планируемых результатов и уровни освоения программы учащимися.

Уровни	Параметры	Показатели
Высокий уровень (80-100%)	Теоретические знания.	Обучающийся глубоко и всесторонне усвоил навыки работы; уверенно, логично, последовательно и грамотно излагает материал; умело обосновывает и аргументирует выдвигаемые им идеи; делает выводы и обобщения; свободно владеет понятиями.

	Практические умения и навыки.	Способен применять практические умения и навыки во время выполнения самостоятельных заданий. Работу выполняет с соблюдением правил техники безопасности, аккуратно, доводит ее до конца. Может оценить результаты выполнения своего задания и дать оценку работы своего товарища.
Средний уровень (50-79%)	Теоретические знания.	Тема раскрыта недостаточно четко и полно, то есть обучающийся освоил проблему, по существу излагает ее, но допускает несущественные ошибки и неточности; слабо аргументирует научные положения; затрудняется в формулировании выводов и обобщений; частично владеет системой понятий.
	Практические умения и навыки.	Владеет базовыми навыками и умениями, но не всегда может выполнить самостоятельное задание, затрудняется и просит помощи педагога. В работе допускает небрежность, делает ошибки, но может устранить их после наводящих вопросов или самостоятельно. Оценить результаты своей деятельности может с подсказкой педагога.
Низкий уровень (меньше 50%)	Теоретические знания.	Обучающийся не усвоил значительной части проблемы, допускает существенные ошибки и неточности при рассмотрении ее; не может аргументировать научные положения; не формулирует выводов и обобщений; не владеет понятийным аппаратом.
	Практические умения и навыки.	Владеет минимальными начальными навыками и умениями. Учащийся способен выполнять каждую операцию только с подсказкой педагога или товарищей. В работе допускает грубые ошибки, не может их найти их даже после указания. Не способен самостоятельно оценить результаты своей работы.

Методические материалы

Для освоения программы используются разнообразные приемы и методы обучения и

воспитания, выбор которых осуществляется с учетом возможностей обучающихся, их возрастных и психофизических особенностей.

Используемые методы обучения, классифицируемые по источнику знаний Н. М. Верзилиным и В. М. Корсунской:

- *наглядные методы*: демонстрации натуральных объектов, изобразительных средств наглядности (таблицы, фильмы, картин, рисунков, схем, шаблонов, образцов, муляжей и моделей объектов);
- *словесные методы*: сюжетный, иллюстративный, информационный рассказ; лекция; объяснение; доказательство; объяснительно-иллюстративная и эвристическая беседа;
- *практические методы*: проведение практических работ;
- *методы мультимедийного обучения*: мультимедийная лекция, виртуальная практическая работа, работа с обучающими компьютерными программами и учебными играми и другие;
- *игровые методы*.

Программа строится на следующих **дидактических принципах** общей педагогики:

- *принцип научности* (отбираемое содержание должно отвечать достижениям науки в соответствующей области знаний);
- *принцип систематичности и последовательности* (последовательное, с учетом логики конкретной науки и интеллектуальных возможностей обучающихся, развертывание содержания знаний, способов деятельности);
- *принцип сознания обучения* (знания становятся достоянием человека в результате самостоятельной сознательной деятельности);
- *принцип активности и самостоятельности*;
- *принцип наглядности*;
- *принцип доступности* (оптимальный для усвоения объем материала, переход от простого к сложному, от известного к неизвестному);
- *принцип основательности* (получение хорошо осознанных, систематизированных, связанных с практикой знаний, освоение умений и навыков);
- *принцип последовательности* (строгая поэтапность выполнения практических заданий и прохождения разделов, их логическая преемственность в процессе осуществления);
- *принцип связи обучения с практической деятельностью, реалиями жизни*;
- *принцип единства образовательных, развивающих и воспитательных функций обучения*.

Список литературы

1. Гуриков С.Р. Основы алгоритмизации и программирования на Python / С.Р.Гуриков. –М.:«Форум», 2018 – 343 с.
2. Доусен М. «Программируем на Python» (Python Programming for the AbsoluteBeginner) /М.Доусен. – СПб: «Питер», 2016. – 416 с.
3. Лутц М. «Изучаем Python», 4 издание, – Пер. с англ. / М. Лутц – СПб: «Символ-Плюс», 2011. – 1280 с.
4. Любанович Б. «Простой Python. Современный стиль программирования» / Б. Любанович. – СПб: «Питер», 2016. – 480 с.
5. Поляков К.Ю. Программирование. Python.C++: учебное пособие / К.Ю.Поляков. М.: Бинوم. Лаборатория знаний, 2019. В 4-х т.
6. Прохоренок Н. «Python 3 и PyQt 5. Разработка приложений» / Н. Прохоренок,В. Дронов. СПб: «БХВ-Петербург», 2019. – 832 с.
7. Саммерфильд М. «Python на практике», пер. А. Слинкин / М.: «ДМК-Пресс»,2014. – 338с.

Электронные образовательные ресурсы

1. Центр онлайн-обучения «Фоксфорд» <http://foxford.ru/>;
2. Яндекс учебник. <https://education.yandex.ru>
3. Образовательная платформа и конструктор онлайн-курсов Stepik.
<https://stepik.org/lesson/3361/step/2?unit=944>
4. "Поколение Python": курс для начинающих.
<https://stepik.org/lesson/265077/step/14?unit=246025c>
5. Всероссийский портал «Дистанционная подготовка по информатике»(informatics.mcsme.ru);
6. Официальная документация языка Python (docs.python.org).