

Аннотация

Основная цель образовательного модуля - привлечь детей к исследовательской и изобретательской деятельности, показать им, что в современном мире область программирования очень интересна и перспективна.

Python – это универсальный современный язык программирования высокого уровня, к преимуществам которого относят высокую производительность программных решений и структурированный, хорошо читаемый код. Синтаксис Python максимально облегчен, что позволяет выучить его за сравнительно короткое время. Развивается быстро, за счет большого количества встроенных библиотек применим во многих IT областях, связанных в том числе с инженерией, производством, автоматизацией и др. На ряду с Си является ведущем языком большинства направлений Национальной технической инициативы (НТИ) – государственной программы мер по поддержке развития в России перспективных отраслей, которые в течение следующих 20 лет могут стать основой мировой экономики.

Python – это текстовый язык программирования. Он универсален, пригоден для создания самых разных программ, от текстовых процессоров до веб-браузеров. Вот несколько причин, почему именно этот язык я предлагаю изучать в рамках занятий: Python – простой и удобный язык. По сравнению со многими другими языками читать и составлять программы на Python совсем не сложно; В Python есть библиотеки готовых процедур для использования в программах. Это позволяет создавать сложные программы быстро; Python используется серьёзными организациями в серьёзных проектах. Например, его используют в Google, Amazon, Yandex, NASA.

Пояснительная записка

Программа «Изучение языка Python. Базовый уровень» составлена на основании Федерального закона об образовании № 273-ФЗ от 29.12.2012 г. с изменениями и дополнениями 2012, 2015 гг., Приказа Министерства просвещения Российской Федерации от 09.11.2018 г. №196 «Порядок организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам», Письма Министерства образования и науки Российской Федерации от 18.11.2015 г. № 09-3242 «Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы)», Концепции дополнительного образования детей, СанПиН 2.4.4.3172-14 «Санитарно-эпидемиологические требования к устройству, содержанию и организации режима работы образовательных организаций дополнительного образования детей», Устава организации и других локальных документов и актов, регламентирующих работу в рамках реализации дополнительной общеобразовательной программы.

Направленность программы: техническая.

Объем освоения программы: 216 часов.

Срок освоения программы: до 1 года.

Форма обучения: очная.

Режим занятий: 2 раза в неделю по 3 часа. Продолжительность одного учебного часа – 45 минут.

Количество обучающихся в группе: 12-14 человек.

Возраст обучающихся: 11 – 13 лет.

Распределение комплектов оборудования и материалов – 1 комплект на 1 обучающегося.

Цель:

Развитие навыков самостоятельного создания компьютерных программ на языке Python,

Задачи:

Образовательные:

Обучение основам моделирования и программирования.

Обучение методам сбора и обработки данных для решения математических задач.

Обучение методам численных расчетов с помощью языка программирования Python.

Формирование навыков работы с языком Python.

Формирование навыков решения прикладных задач посредством программирования.

Формирование навыков самостоятельного обучения и поиска решения возникшей в процессе работы проблеме.

Развивающие:

Развитие абстрактного, логического и алгоритмического мышления.

Развитие операций мышления (анализ, синтез).

Формирование способности и готовности к использованию ИКТ в целях обучения и развития.

Развитие творческих способностей ребенка.

Воспитательные:

Воспитание способности к сотрудничеству и коммуникации.

Формирование общей культуры личности.

Воспитание способности к самоорганизации и планированию своей деятельности.

Содержание программы
Учебный тематический план

Таблица № 1

№ п/п	Название раздела, темы	Количество часов		
		Всего	Теория	Практика
1	Вводное занятие. Знакомство учащихся. Вводный инструктаж по ТБ.	3	1	2
2	Раздел 1. Знакомство с языком Python	12	5	7
3	Раздел 2. Переменные и выражения	18	6	12
4	Раздел 3. Условные предложения	33	8	22
5	Раздел 4. Циклы	51	9	37
6	Раздел 5. Функции	27	6	18
7	Раздел 6. Строки - последовательности символов	18	3	9
8	Раздел 7. Сложные типы данных	27	7	14
9	Раздел 8. Стиль программирования и отладка программ	27	6	21
10	Выходное тестирование	3	0	3
11	Подведение итогов курса	3	0	3
Всего:		216	72	144

№ п/п	Наименование тем	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
	Вводное занятие/тестирование. Знакомство учащихся Инструктаж по технике безопасности	3	1	2	Игра, Опрос
1.	Раздел 1. Знакомство с языком Python	12	5	7	Опрос, решение практических задач
1.1	Тема 1. Общие сведения о языке Практическая работа: Установка программы Python	3	2	1	
1.2.	Тема 2. Режимы работы Практическая работа: Режимы работы с Python	9	3	6	
2.	Раздел 2. Переменные и выражения	18	6	12	Педагогическое наблюдение, решение практических задач
2.1	Тема 3. Переменные Практическая работа: Работа со справочной системой	3	1	2	
2.2	Практическая работа: Переменные	3	-	3	
2.3	Тема 4. Выражения Практическая работа: Выражения	3	1	2	
2.4	Тема 5. Ввод и вывод	3	1	2	

2.5	Тема 6. Задачи на элементарные действия с числами Практическая работа 2.5. Задачи на элементарные действия с числами	6	3	3	
3	Раздел 3. Условные предложения	33	8	22	
3.1	Тема 7. Логические выражения и операторы. Практическая работа: Логические выражения	3	1	2	Тестирование, Решение практических задач
3.2	Тема 8. Условный оператор Практическая работа: «Условный оператор»	6	2	4	
3.3	Тема 9. Множественное ветвление Практическая работа: Множественное ветвление	6	2	4	
3.4	Тема 10. Реализация ветвления в языке Python. Практическая работа: «Условные операторы»	3	1	2	
3.5	Самостоятельная работа № 1 по теме «Условные операторы».	9	2	7	

3.6	Тема 11. Зачетная работа № 1. "Составление программ с ветвлением".	4	-	4	
3.7	Тест № 3. "Условные операторы".	2	-	2	
4	Раздел 4. Циклы	51	9	37	
4.1.	Тема 12. Оператор цикла с условием Практическая работа "Числа Фибоначчи"	4	1	3	Решение практических задач, творческая работа
4.2.	Тема 13. Оператор цикла for Практическая работа Решение задачи с циклом for.	4	1	3	
4.3.	Тема 14. Вложенные циклы Практическая работа: Реализация циклических алгоритмов	4	1	3	
4.4.	Тема 15. Случайные числа Практическая работа: Случайные числа	4	1	3	
4.5.	Тема 16. Примеры решения задач с циклом Практическая работа: Решение задач с циклом.	6	1	5	

4.6	Самостоятельная работа № 2 "Составление программ с циклом"	10	2	8	
4.7	Тест № 4. Циклы	2	-	2	
4.8	Тема 17. Творческая работа № 1. "Циклы"	12	2	10	
5	Раздел 5. Функции	27	6	18	Тестирование, решение практических задач
5.1.	Тема 18. Создание функций Практическая работа Создание функций	2	1	2	
5.2.	Тема 19. Локальные переменные Практическая работа Локальные переменные	2	1	2	
5.3	Тема 20. Примеры решения задач с использованием функций Практическая работа Решение задач с использованием функций	2	1	2	
5.4	Самостоятельная работа № 3 по теме "Функции"	10	2	8	
5.5	Тема 21. Рекурсивные функции Практическая работа 5.4. Рекурсивные функции	2	1	1	
5.6	Тест № 5. Функции	2	-	3	

6	Раздел 6. Строки - последовательности символов	18	3	9	Решение практических задач
6.1.	Тема 22. Строки Практическая работа: Строки	2	1	1	
6.2.	Тема 23. Срезы строк	2	1	1	
6.3	Тема 24. Примеры решения задач со строками Практическая работа: Решение задач со строками.	6	1	5	
7	Раздел 7. Сложные типы данных	27	7	14	Тестирование, Решение практических задач
7.1.	Тема 25. Списки	3	1	2	
7.2	Тема 26. Срезы списков	2	1	1	
7.3	Тема 27. Списки: примеры решения задач Практическая работа 7.2. Решение задач со списками	6	1	5	
7.4	Тема 28. Матрицы	2	1	1	
7.5	Тест № 7. Списки	2	-	2	
7.6	Тема 29. Кортежи	2	1	1	
7.7	Тема 30. Словари	2	1	1	
7.8	Тема 31. Множества в языке Python	2	1	1	

8	Раздел 8. Стил программирования и отладка программ	27	6	21	Решение практических задач
8.1	Тема 32. Стил программирования	2	1	1	
8.2	Тема 33. Отладка программ	3	1	2	
8.3	Тема 34. Творческий проект	21	3	18	Защита презентации творческого проекта
8.3	Тема 35. Итоги	1	1	-	Дискуссия
	ВСЕГО	216	72	144	

Содержание учебного плана

Вводное занятие (2 часа).

Теория: Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Игр-тренинг на командообразование.

Раздел 1. Знакомство с языком Python (12 часов)

Теория: Общие сведения о языке Python. Установка Python на компьютер. Режимы работы Python. Что такое программа. Первая программа. Структура программы на языке Python. Комментарии.

Практическая работа 1.1. Установка программы Python Практическая работа 1.2. Режимы работы с Python Тест № 1. Знакомство с языком Python

Учащиеся должны знать / понимать:

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

Учащиеся должны уметь:

- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе.

Раздел 2. Переменные и выражения (18 часов)

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания.

Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Матричные функции.

Композиция.

Ввод и вывод. Ввод данных с клавиатуры. Вывод данных на экран. Пример скрипта, использующего ввод и вывод данных. Задачи на элементарные действия с числами. Решение задач на элементарные действия с числами.

Практическая работа 2.1. Работа со справочной системой Практическая работа 2.2.

Переменные

Практическая работа 2.3. Выражения

Практическая работа 2.5. Задачи на элементарные действия с числами Тест № 2.

Выражения и операции.

Учащиеся должны знать / понимать:

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

Раздел 3. Условные предложения (33 часа)

Теория: Логический тип данных. Логические выражения и операторы. Сложные условные выражения (логические операции and, or, not). Условный оператор. Альтернативное выполнение. Примеры решения задач с условным оператором. Множественное ветвление. Реализация ветвления в языке Python.

Практическая работа 3.1. Логические выражения Практическая работа 3.2. "Условный оператор" Практическая работа 3.3. Множественное ветвление Практическая работа 3.4. "Условные операторы"

Самостоятельная работа № 1. Решение задач по теме "Условные операторы". Зачетная работа № 1. "Составление программ с ветвлением".

Тест № 3. "Условные операторы".

Учащиеся должны знать / понимать:

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

Раздел 4. Циклы (51 час)

Теория: Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием. Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while. Обновление переменной. Краткая форма записи обновления. Примеры использования циклов.

Оператор цикла с параметром for. Операторы управления циклом. Пример задачи с использованием цикла for. Вложенные циклы. Циклы в циклах. Случайные числа. Функция randrange. Функция random. Примеры решения задач с циклом.

Практическая работа 4.1. "Числа Фибоначчи" Практическая работа 4.2. Решение задачи с циклом for.

Практическая работа 4.3. Реализация циклических алгоритмов Практическая работа 4.4. Случайные числа

Практическая работа 4.5. Решение задач с циклом. Самостоятельная работа № 2 "Составление программ с циклом"

Тест № 4. Циклы

Творческая работа № 1. "Циклы"

Учащиеся должны знать / понимать:

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

Учащиеся должны уметь:

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

Раздел 5. Функции (27 часов)

Теория: Создание функций. Параметры и аргументы. Локальные и глобальные переменные.

Поток выполнения. Функции, возвращающие результат. Анонимные функции, инструкция lambda. Примеры решения задач с использованием функций.

Рекурсивные функции. Вычисление факториала. Числа Фибоначчи. Практическая работа 5.1. Создание функций

Практическая работа 5.2. Локальные переменные

Практическая работа 5.3. Решение задач с использованием функций Практическая работа

5.4. Рекурсивные функции

Самостоятельная работа № 3 по теме "Функции" Тест № 5. Функции

Учащиеся должны знать / понимать:

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

Учащиеся должны уметь:

- создавать и использовать функции;
- использовать механизм параметров для передачи значений.

Раздел 6. Строки - последовательности символов (18 часов)

Теория: Составной тип данных - строка. Доступ по индексу. Длина строки и отрицательные индексы. Преобразование типов. Применение цикла для обхода строки.

Срезы строк. Строки нельзя изменить. Сравнение строк. Оператор in. Модуль string. Операторы для всех типов последовательностей (строки, списки, кортежи). Примеры решения задач со строками.

Практическая работа 6.1. Строки

Практическая работа 6.2. Решение задач со строками.

Учащиеся должны знать / понимать:

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

Учащиеся должны уметь:

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

Раздел 7. Сложные типы данных (27 часов)

Теория: Списки. Тип список (list). Индексы. Обход списка. Проверка вхождения в список. Добавление в список. Суммирование или изменение списка. Операторы для списков. Срезы

списков. Удаление списка. Клонирование списков. Списочные параметры. Функция range.
Списки: примеры решения задач.

Матрицы. Вложенные списки. Матрицы. Строки и списки. Генераторы списков в Python.
Кортежи. Присваивание кортежей. Кортежи как возвращаемые значения

Введение в словари. Тип словарь (dict). Словарные операции. Словарные методы.

Множества в языке Python. Множества. Множественный тип данных. Описание множеств.
Операции, допустимые над множествами: объединение, пересечение, разность, включение.
Оператор определения принадлежности элемента множеству.

Практическая работа 7.1. Списки.

Практическая работа 7.2. Решение задач со списками. Тест № 7. Списки

Учащиеся должны знать / понимать:

- сложные типы данных;
- способ описания списка;
- способ доступа к элементам списка;
- способ описания кортежа;
- способ описания словаря;
- операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.

Учащиеся должны уметь:

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

Раздел 8. Стиль программирования и отладка программ (27 часов)

Стиль программирования. Отладка программ.

Зачет по курсу «Программирование на языке Python»

Учащиеся должны знать / понимать:

- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

Учащиеся должны уметь:

- определять вид ошибок и находить ошибки в программе.
- выполнять тестирование и отладку программ.

Планируемые результаты освоения программы

В рамках курса «Программирование на языке Python» учащиеся овладеют следующими знаниями, умениями и способами деятельности:

- умеют составлять алгоритмы для решения задач;
- умеют реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- владеют основными навыками программирования на языке Python;
- умеют отлаживать и тестировать программы, написанные на языке Python.

Способы оценивания уровня достижений учащихся

Предметом диагностики и контроля являются внешние образовательные продукты учащихся (созданные блок-схемы, программы), а также их внутренние личностные качества (освоенные способы деятельности, знания, умения), которые относятся к целям и задачам курса.

Качество внешней образовательной продукции желательно оценивать по следующим параметрам:

- алгоритм должен быть оптимальным по скорости выполнения и максимально простым в реализации на языке программирования;
 - программа должна выполнять поставленные задачи;
 - по степени «читаемости кода» (должны быть соблюдены отступы, обязательное наличие комментариев к коду программы и т. д.).
- Проверка достигаемых учащимися результатов производится в следующих формах:
- текущий рефлексивный самоанализ, контроль и самооценка учащимися выполняемых заданий;
 - текущая диагностика и оценка педагогом деятельности учащихся;
 - итоговая оценка деятельности и образовательной продукции ученика в соответствии с его индивидуальными и возрастными особенностями;

Итоговый контроль проводится в конце всего курса. Он состоит из тестирования и решения

практической задачи, защиты творческого проекта.

Прогнозируемые результаты

Предметные результаты:

- знание языка Python;
- выполнение простейшей программы в интерактивной среде;
- умение различать различные виды циклов;
- понимание принципов структурного программирования;
- умение решать простейшие олимпиадные задачи.

Метапредметные результаты:

- способность сознательно организовывать и регулировать свою деятельность – учебную, общественную и др.;
- владение умениями работать с учебной и внешкольной информацией (анализировать и обобщать факты, составлять планирование своей деятельности, тезисы, конспект, формулировать и обосновывать выводы и т.д.), использовать современные источники информации, в том числе материалы на электронных носителях;
- способность решать творческие задачи, представлять результаты своей деятельности в различных формах (сообщение, доклад, реферат, презентация и др.).

Личностные результаты:

- освоение гуманистических традиций и ценностей современного общества;
- осмысление социально-нравственного опыта предшествующих поколений, способность к определению своей позиции и ответственному поведению в современном обществе;
- понимание многообразия мира.

**КОМПЛЕКС ОРГАНИЗАЦИОННО-ПЕДАГОГИЧЕСКИХ УСЛОВИЙ
ДОПОЛНИТЕЛЬНОЙ ОБЩЕОБРАЗОВАТЕЛЬНОЙ ОБЩЕРАЗВИВАЮЩЕЙ
ПРОГРАММЫ:**

КАЛЕНДАРНЫЙ УЧЕБНЫЙ ГРАФИК

Календарный учебный график прилагается к журналу учета работы объединения.

ФОРМЫ АТТЕСТАЦИИ И ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Для отслеживания результативности образовательного процесса используются следующие виды контроля:

- входящий (проводится в начале учебного года для выявления уровня знаний детей),
- текущий (проводится в течение всего учебного года с целью определения степени усвоения учебного материала и подбора наиболее эффективных методов и средств обучения),
- промежуточный (проводится в конце полугодия, четверти или темы для определения результатов обучения),
- итоговый (проводится в конце учебного года с целью определения уровня развития детей, степени освоения образовательной программы).

Итоговая работа

Итоговый контроль обучающихся проводится по результатам выполнения практических заданий и защиты проектов (правила выбора темы и пример проекта представлены в приложение №1)

МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Формы учебной деятельности:

- практическое занятие;
- занятие с творческим заданием;
- викторина;
- выставка;
- экскурсия.

Виды учебной деятельности:

- решение поставленных задач;
- просмотр и обсуждение учебных фильмов, презентаций, роликов;
- объяснение и интерпретация наблюдаемых явлений;

- анализ проблемных учебных ситуаций;
- построение гипотезы на основе анализа имеющихся данных;
- проведение исследовательского эксперимента.
- поиск необходимой информации в учебной и справочной литературе;
- выполнение практических работ;
- подготовка выступлений и докладов с использованием разнообразных источников информации;

информации;

- публичное выступление.

С целью повышения доступности и качества обучения программой предусмотрено сочетание традиционных и дистанционных образовательных технологий в соответствии с интересами и возможностями обучающихся, их способностями и потребностями.

А также организация использования дистанционных образовательных технологий в образовательном процессе в дни невозможности посещения занятий обучающимися по неблагоприятным погодным условиям, по болезни или в период карантина, с целью установления единых подходов к деятельности детского технопарка «Кванториум Сампо», обеспечения усвоения обучающимися обязательного минимума содержания образовательных программ и регулирования организации обучения с использованием дистанционных образовательных технологий.

Образовательный процесс с использованием дистанционных образовательных технологий предусматривает значительную долю самостоятельных занятий обучающихся, возможность получения консультаций педагога, а также выполнение творческих заданий.

В процессе проведения обучения в дистанционном режиме используются:

- *электронная почта*
- *пересылка данных*
- *гипертекстовые среды*
- *ресурсы мировой сети Интернет*
- *видеоконференции*

Правила выбора темы проекта

Способы решения проблем начинающими исследователями во многом зависят от выбранной темы. Надо помочь детям найти все пути, ведущие к достижению цели, выделить общепринятые, общеизвестные и нестандартные, альтернативные; сделать выбор, оценив эффективность каждого способа.

Правило 1. Тема должна быть интересна ребенку, должна увлекать его. Исследовательская работа эффективна только на добровольной основе. Тема, навязанная ученику, какой бы важной она ни казалась взрослым, не даст должного эффекта.

Правило 2. Тема должна быть выполнима, решение ее должно быть полезно участникам исследования. Натолкнуть ребенка на ту идею, в которой он максимально реализуется как исследователь, раскроет лучшие стороны своего интеллекта, получит новые полезные знания, умения и навыки, – сложная, но необходимая задача для педагога.

Правило 3. Тема должна быть оригинальной с элементами неожиданности, необычности. Оригинальность следует понимать как способность нестандартно смотреть на традиционные предметы и явления.

Правило 4. Тема должна быть такой, чтобы работа могла быть выполнена относительно быстро. Способность долго концентрировать собственное внимание на одном объекте, т. е. долговременно, целеустремленно работать в одном направлении, у школьника ограничена.

Правило 5. Тема должна быть доступной. Она должна соответствовать возрастным особенностям детей. Это касается не только выбора темы исследования, но и формулировки и отбора материала для ее решения. Одна и та же проблема может решаться разными возрастными группами на различных этапах обучения.

Правило 6. Сочетание желаний и возможностей. Выбирая тему, педагог должен учесть наличие требуемых средств и материалов – исследовательской базы. Ее отсутствие, невозможность собрать необходимые данные обычно приводят к поверхностному решению, порождают "пустословие". Это мешает развитию критического мышления, основанного на доказательном исследовании и надежных знаниях.

Правило 7. С выбором темы не стоит затягивать. Большинство учащихся не имеют постоянных пристрастий, их интересы ситуативны. Поэтому, выбирая тему, действовать следует быстро, пока интерес не угас.