

Муниципальное автономное учреждение дополнительного образования
«Центр дополнительного образования» г. Мирный
муниципального образования «Мирнинский район» Республики Саха (Якутия)

Принята на заседании
педагогического совета
МАУ ДО «ЦДО» г. Мирный

Протокол №1 от «09» сентября 2022г.



УТВЕРЖДАЮ
Директор МАУ ДО ЦДО г. Мирный

И.Ю. Федоров
Приказ №203 от «09» сентября 2022г.

**Дополнительная общеобразовательная
общеразвивающая программа
технической направленности
«Программирование на Python» (старшие)**

Тип программы: модифицированная
Срок реализации: 1 год
Возраст обучающихся: 14-18 лет
Уровень: базовый

Составитель:
Кондаков Михаил Алексеевич,
педагог дополнительного образования

Раздел 1. Комплекс основных характеристик программы

1.1. Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Программирование на Python» имеет техническую направленность.

Дополнительная общеобразовательная общеразвивающая программа составлена с учетом:

Федерального Закона Российской Федерации от 29.12.2012 273-ФЗ «Об образовании в Российской Федерации»;

Приказ Министерства просвещения Российской Федерации "Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам" от 09.11.2018 №196;

Постановление Государственного санитарного врача РФ от 28.09.2020г. "Об утверждении санитарных правил СП-2.4.3648-20 "Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и м

о Методические рекомендации по проектированию дополнительных общеразвивающих программ (включая разноуровневые программы) (Приложение к письму Департамента государственной политики в сфере воспитания детей и молодежи Министерства образования и науки РФ от 18.11.2015 № 09-3242).

е Приказ Министерства просвещения России от 03.09.2019 N 467 (ред. от 02.02.2021) «Об утверждении Целевой модели развития региональных систем дополнительного образования детей».

С В рамках реализации федерального проекта «Цифровая образовательная среда» национального проекта «Образование» в Детском технопарке г. Мирный функционирует Центр цифрового образования детей «IT-куб». Целью создания ЦЦОД «IT-куб» является — создание инфраструктурной площадки для приобщения учащихся к инновационной, практико-ориентированной деятельности в сфере информационных технологий, робототехники и IT-инжиниринга. Данный проект нацелен на популяризацию технических профессий, увеличение охвата учащихся IT-технологиями и направлен на раннюю профессиональную ориентацию.

б Дополнительная общеобразовательная программа «Программирование на Python (старшие)» в соответствии с утвержденным Федеральным оператором перечнем направлений, относится к обязательному направлению ЦЦОД «IT-куб».

- Кабинет отремонтирован, брендирован, оснащен высокотехнологичным оборудованием, соответствует требованиям Санитарных правил: просторный, светлый, с достаточным количеством розеток и высокоскоростным интернетом.

; **Актуальность**

Язык программирования Python на сегодняшний день является наиболее популярной высокоуровневой средой международного олимпиадного программирования. Python – мощный и простой для изучения язык программирования. В нем представлены проработанные высокоуровневые структуры данных и простой, но эффективный подход к объектно-ориентированному программированию. Данная программа направлена на расширение знаний обучающихся в направлении изучения языков программирования, в частности, языка программирования Python.

Методы, применяемые в процессе обучения, такие как проблемное обучение, проектная деятельность, способствуют формированию мотивации учащихся к углубленному изучению программирования, как одной из компьютерных наук. У детей формируется познавательный интерес, самостоятельность мышления, стремление к самопознанию.

Отличительная особенность программы состоит в том, что она позволяет привлечь детей среднего и старшего школьного возраста к изучению программирования при помощи языка Python, так как он обладает следующими достоинствами:

- Python – это текстовый язык программирования. Он универсален, пригоден для создания самых разных программ, от текстовых процессоров до веб-браузеров;

- 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 своих программах. Это позволяет создавать сложные программы быстро;

Новизна заключается в том, что Python дает более широкие возможности в области программирования, чем другие языки программирования, которые входят в школьный курс информатики. На языке Python можно легко и быстро создавать простые компьютерные игры и программировать роботов. Этот язык быстрее и легче усваивается. Многие мировые компании такие, как Intel, Cisco, Hewlett-Packard, используют этот язык при реализации своих проектов. Крупнейшие интернет-ресурсы такие, как Google, YouTube, также разработаны с помощью языка программирования Python.

Адресат дополнительной общеобразовательной программы

Программа составлена с учетом поло-возрастных и индивидуально-психологических, физических особенностей обучающихся и рассчитана на обучающихся 14-18 лет соответствует направлениям деятельности данного объединения и относится к базовому уровню сложности.

Студию могут посещать как мальчики, так и девочки. Количество обучающихся не менее 8 человек в группе.

Занятия проводятся:

1 год обучения – 2 занятия в неделю по 2 часа. Всего 144 часа.

Продолжительность одного академического часа - 45 мин. Перерыв между учебными занятиями – 10 минут. Общее количество часов в неделю – 4 часов.

Формы реализации: Занятия проводятся в заочной форме обучения с использованием дистанционных технологий и использованием сетевой формы.

Дистанционное обучение

С учетом санитарно-эпидемиологической ситуации при необходимости возможна реализация программы в заочной форме обучения с использованием дистанционных технологий, без использования сетевой формы.

Опосредованное осуществление взаимодействия педагога с обучающимися может быть организовано при подготовке к участию в конкурсах на заключительных этапах; для обучающихся с высокой степенью успешности в освоении программ; для обучающихся, пропускающих учебные занятия по уважительной причине (болезнь и др.); в период отмены (приостановки) занятий в очной (контактной) форме. В обучении с применением ЭО и ДОТ могут использоваться следующие организационные формы учебной деятельности:

- лекция;
- презентация;
- онлайн-беседа;
- упражнения;
- практическое занятие;
- контрольная работа;
- самостоятельная работа.

Электронные образовательные ресурсы, используемые при применении дистанционных образовательных технологий:

- Zoom <https://zoom.us/> — эта платформа для конференций дает возможность бесплатно организовывать встречи до 100 участников;

- Google Класс Google Classroom – бесплатный сервис для школ, некоммерческих организаций и всех, у кого есть личный аккаунт Google. Он упрощает диалог учащихся и преподавателей. Этот сервис позволяет экономить время при создании курсов, рассылке домашних заданий, общении с учащимися и организации учебного процесса;

- Яндекс.Диск — облачный сервис, позволяющий пользователям хранить свои данные на серверах в «облаке» и передавать их другим пользователям в Интернете.

Педагогическая целесообразность: заключается в том, что данная программа позволит выявить заинтересованных обучающихся, проявивших интерес к знаниям, оказать им помощь в формировании устойчивого интереса к программированию на языке Python. В процессе изучения Python обучающиеся научатся программировать на языке будущего, это повысит уровень логического мышления, а также аналитический склад ума.

1.2. Цель и задачи программы

Цель: Формирование творческой личности, обладающей информационными компетенциями, владеющей базовыми понятиями теории алгоритмов, умеющей разрабатывать эффективные алгоритмы и реализовывать их в виде программы, написанной на языке программирования Python.

Задачи

Образовательные:

- познакомить с понятиями алгоритма, вычислимой функции, языка программирования;
- научить составлять и читать блок-схемы;
- сформировать навыки выполнения технологической цепочки разработки программ средствами языка программирования Python;
- объяснить основные конструкции языка программирования Python, позволяющие работать с простыми и составными типами данных (строками, списками, кортежами, словарями, множествами);
- научить применять функции при написании программ на языке программирования Python;
- научить отлаживать и тестировать программы, делать выводы о работе этих программ.

Воспитательные:

- формировать правильный методологический подход к познавательной и практической деятельности;
- формирование мотивации обучающихся к изобретательству, созданию собственных программных реализаций;
- способствовать стремлению к овладению техникой исследования;
- воспитывать трудолюбие, инициативность и настойчивость в преодолении трудностей;
- стремление к получению качественного законченного результата в проектной деятельности;
- развивать навыки сознательного и рационального использования компьютера в своей учебной, а впоследствии и в профессиональной деятельности

Развивающие:

- развить познавательные процессы (внимание, восприятие, логическое мышление, память),
- развить креативность,
- развить способности к самореализации.

1.3. Содержание программы

Учебно-тематическое планирование

Основная форма работы теоретической части – лекционные занятия. Практические задания планируется выполнять индивидуально, в парах и в малых группах. Занятия проводятся в виде бесед, семинаров, лекций: для наглядности изучаемого материала используется различный мультимедийный материал – презентации.

№ п/п	Наименование тем	Количество часов			Формы аттестации/ контроля
		Всего	Теория	Практика	
1.	Раздел 1. Вводное занятие	2	2	-	Теория
1.1	Инструктаж по технике безопасности в компьютерном классе	2	2	-	
2.	Раздел 2. Знакомство с языком Python	6	2	4	Тестирование, решение практических задач
2.1	Общие сведения о языке Практическая работа: Установка программы Python	2	1	1	
2.2.	Режимы работы Практическая работа: Режимы работы с Python	2	1	1	
2.3	Тест. Знакомство с языком Python	2		2	
3.	Раздел 3. Переменные и выражения	10	3	7	Тестирование, решение практических задач
3.1	Переменные Практическая работа: Работа со справочной системой	2	1	1	
3.2	Практическая работа: Переменные	2		2	
3.3	Выражения Практическая работа: Выражения	1		1	
3.4	Ввод и вывод	1	1		
3.5	Задачи на элементарные действия с числами Практическая работа. Задачи на элементарные действия с числами	2	1	1	
3.6	Тест. Выражения и операции.	2		2	
4.	Раздел 4. Условные предложения	28	6	22	Тестирование, Решение практических задач
4.1	Логические выражения и операторы. Практическая работа: Логические выражения	2	1	1	
4.2	Условный оператор Практическая работа: «Условный оператор»	4	1	3	
4.3	Множественное ветвление Практическая работа: Множественное ветвление	4	1	3	
4.4	Реализация ветвления в языке Python. Практическая работа: «Условные операторы»	2	1	1	
4.4	Самостоятельная работа по	10	2	8	

	теме «Условные операторы».				
4.5	Зачетная работа № 1. "Составление программ с ветвлением".	4		4	
4.6	Тест. "Условные операторы".	2		2	
5.	Раздел 5. Циклы	36	14	22	
5.1	Оператор цикла с условием Практическая работа "Числа Фибоначчи"	4	2	2	Тестирование, решение практических задач, творческая работа
5.2	Оператор цикла for Практическая работа Решение задачи с циклом for.	4	2	2	
5.3	Вложенные циклы Практическая работа: Реализация циклических алгоритмов	2	1	1	
5.4	Случайные числа Практическая работа: Случайные числа	2	1	1	
5.5	Примеры решения задач с циклом Практическая работа: Решение задач с циклом.	4	2	2	
5.6	Самостоятельная работа "Составление программ с циклом"	6	2	4	
5.7	Тест. Циклы	2		2	
5.8	Творческая работа № 1. "Циклы"	12	4	8	
6.	Раздел 6. Функции	20	6	14	
6.1	Создание функций Практическая работа Создание функций	2	1	1	Тестирование, решение практических задач
6.2	Локальные переменные Практическая работа Локальные переменные	2	1	1	
6.3	Примеры решения задач с использованием функций Практическая работа. Решение задач с использованием функций	2	1	1	
6.4	Самостоятельная работа по теме "Функции"	10	2	8	
6.5	Рекурсивные функции Практическая работа. Рекурсивные функции	2	1	1	
6.6	Тест. Функции	2		2	
7.	Раздел 7. Строки - последовательности символов	10	4	6	Решение практических задач
7.1	Строки	2	1	1	

	Практическая работа: Строки				
7.2	Срезы строк	2	1	1	
7.3	Примеры решения задач со строками Практическая работа: Решение задач со строками.	6	2	4	
8.	Раздел 8. Сложные типы данных	20	10	10	
8.1	Списки Практическая работа: Списки	2	1	1	Тестирование, Решение практических задач
8.2	Срезы списков	2	1	1	
8.3	Списки: примеры решения задач Практическая работа. Решение задач со списками	6	2	4	
8.4	Матрицы	2	1	1	
8.5	Тест. Списки	2		2	
8.6	Кортежи	2	2		
8.7	Введение в словари	2	2		
8.8	Множества в языке Python	2	1	1	
9.	Раздел 9. Стиль программирования и отладка программ	12	7	5	
9.1	Стиль программирования	4	2	2	Решение практических задач, презентация проекта, рефлексия.
9.2	Отладка программ	4	2	2	
9.3	Зачет по курсу «Программирование на языке Python»	2	1	1	
9.4	Что дальше?	2	2		
	ВСЕГО	144	53	91	

Содержание учебного плана 1 год обучения

Раздел 1. Вводное занятие (2 часа).

Теория: Знакомство с планом работы объединения, инструктаж по ТБ. Практика: Опрос. Тренинг на командо-образование.

Раздел 2. Знакомство с языком Python (6 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 3. Переменные и выражения (10 часов)

Теория: Типы данных. Преобразование типов. Переменные. Оператор присваивания. Имена переменных и ключевые слова.

Выражения. Операции. Порядок выполнения операций. Математические 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 4. Условные предложения (28 часов)

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 5. Циклы (36 часов)

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 randint. 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 6. Функции (20 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 7. Строки - последовательности символов (10 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 8. Сложные типы данных (20 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 9. Стиль программирования и отладка программ (12 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

1.4. Планируемые результаты

В рамках курса «Программирование на Python» учащиеся овладевают следующими знаниями, умениями и способами 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.

Способы оценивания уровня достижений учащихся

Предметом диагностики и контроля в курсе «Программирование на Python» являются внешние образовательные продукты обучающихся (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.

Раздел 2. Комплекс организационно-педагогических условий

2.1. Календарный учебный график

Календарный учебный график

(общий)

Год обучения (уровень)	Дата начала занятий	Дата окончания занятий	Количество учебных недель	Количество учебных дней	Количество учебных часов	Режим занятий	Каникулы зимние	Каникулы летние
1 год обучения (базовый/продвинутой)	10 сентября 2022г.	31 мая 2023г.	36	72	144, 4 часа в нед.	2 раза в нед. по 2 часа	01 – 08 января	01 июня – 31 августа

Календарный учебный график

(для 1-ой группы по программе «Программирование на Python» 1 год обучения)

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1	Сентябрь	12	11:45-12:30	Введение	2	Инструктаж по технике безопасности в компьютерном классе	Каб.№8	Опрос
2		15	11:45-12:30	Теория	2	Общие сведения о языке	Каб.№8	Опрос
3		19	11:45-12:30	Теория	2	Режимы работы	Каб.№8	Опрос
4		22	11:45-12:30	Практика	2	Знакомство с языком Python	Каб.№8	Решение задачи
5		26	11:45-12:30	Теория	2	Переменные	Каб.№8	Решение задачи
6		29	11:45-12:30	Теория	2	Выражения	Каб.№8	Тест
8	Октябрь	3	11:45-12:30	Теория	2	Ввод и вывод	Каб.№8	Тест
9		6	11:45-12:30	Теория	2	Выражения и операции.	Каб.№8	Решение задачи
10		10	11:45-12:30	Теория	2	Логические выражения и операторы.	Каб.№8	Решение задачи
11		13	11:45-12:30	Теория	2	Условный оператор	Каб.№8	Тест
12		17	11:45-12:30	Теория	2	Условный оператор Решение задачи.	Каб.№8	Решение задачи
13		20	11:45-12:30	Теория	2	Множественное ветвление	Каб.№8	Решение задачи
14		24	11:45-12:30	Теория	2	Множественное ветвление тест.	Каб.№8	Тест
15		27	11:45-12:30	Теория	2	Реализация ветвления в языке Python.	Каб.№8	Решение задачи
16	Ноябрь	3	11:45-12:30	Теория	2	Реализация ветвления в языке Python на практике	Каб.№8	Решение задач
17		4	11:45-12:30	Теория	2	"Составление программ с ветвлением"	Каб.№8	Решение задачи
18		10	11:45-12:30	Теория	2	Зачетная работа. "Составление программ с ветвлением"	Каб.№8	Тест

19		11	11:45-12:30	Теория	2	"Условные операторы".	Каб.№8	Тест
20		17	11:45-12:30	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
21		18	11:45-12:30	Теория	2		Каб.№8	Решение задачи
22		24	11:45-12:30	Теория	2	Оператор цикла for	Каб.№8	Тест
23		25	11:45-12:30	Теория	2	Оператор цикла for закрепление темы	Каб.№8	Решение задачи
24		30	11:45-12:30	Теория	2	Вложенные циклы	Каб.№8	Решение задач
25	Декабрь	2	11:45-12:30	Теория	2	Вложенные циклы закрепление темы	Каб.№8	Опрос
26		8	11:45-12:30	Теория	2	Случайные числа	Каб.№8	Опрос
27		9	11:45-12:30	Теория	2	Случайные числа тест	Каб.№8	Опрос
28		15	11:45-12:30	Теория	2	Самостоятельная работа. "Составление программ с циклом"	Каб.№8	Решение задачи
29		16	11:45-12:30	Теория	2	Циклы	Каб.№8	Опрос
30		22	11:45-12:30	Теория	2	Циклы решение задач.	Каб.№8	
31		23	11:45-12:30	Теория	2	Примеры решения задач с циклом	Каб.№8	Решение задачи
32		29	11:45-12:30	Теория	2	Творческая работа. "Циклы"	Каб.№8	Решение задачи
33		30	11:45-12:30	Теория	2	Цикл тест	Каб.№8	Тест
34		Январь	11	11:45-12:30	Теория	2	Закрепление темы «Цикл»	Каб.№8
35	12		11:45-12:30	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
36	13		11:45-12:30	Теория	2	Цикл	Каб.№8	Решение задачи
37	19		11:45-12:30	Теория	2	Условия	Каб.№8	Опрос
38	23		11:45-12:30	Теория	2	Создание функций	Каб.№8	Опрос
39	28		11:45-12:30	Теория	2	Создание функций закрепление информации	Каб.№8	Опрос
40	30		11:45-12:30	Теория	2	Создание функций тестовое задание	Каб.№8	Тест
41	Февраль		6	11:45-12:30	Теория	2	Практическая работа Создание функций	Каб.№8
42		13	11:45-12:30	Теория	2	Локальные переменные	Каб.№8	опрос
43		16	11:45-12:30	Теория	2	Локальные переменные Тестовые задания	Каб.№8	Тест
44		20	11:45-12:30	Теория	2	Примеры решения задач с использованием функций	Каб.№8	Тест

45		23	11:45-12:30	Теория	2	Практическая работа Решение задач с использованием функций	Каб.№8	Решение задачи
46		27	11:45-12:30	Теория	2	Самостоятельная работа по теме "Функции"	Каб.№8	Тест
47	Март	1	11:45-12:30	Теория	2	Рекурсивные функции	Каб.№8	Тест
48		3	11:45-12:30	Теория	2	Рекурсивные функции Решение задачи	Каб.№8	Тест
49		6	11:45-12:30	Теория	2	Практическая работа Рекурсивные функции	Каб.№8	Решение задачи
50		9	11:45-12:30	Теория	2	Тест Функции	Каб.№8	Тест
51		13	11:45-12:30	Теория	2	Строки	Каб.№8	Опрос
52		16	11:45-12:30	Теория	2	Строки продолжение	Каб.№8	Опрос
53		23	11:45-12:30	Теория	2	Практическая работа: Строки	Каб.№8	Решение задачи
54		27	11:45-12:30	Теория	2	Строки Решение задач	Каб.№8	Решение задачи
55		30	11:45-12:30	Теория	2	Срезы строк	Каб.№8	Опрос
56		Апрель	3	11:45-12:30	Теория	2	Срезы строк Решение задач	Каб.№8
57	6		11:45-12:30	Теория	2	Примеры решения задач со строками	Каб.№8	Решение задачи
58	10		11:45-12:30	Теория	2	Практическая работа: Решение задач со строками.	Каб.№8	Решение задачи
59	13		11:45-12:30	Теория	2	Списки	Каб.№8	Опрос
60	17		11:45-12:30	Теория	2	Практическая работа: Списки	Каб.№8	Решение задачи
61	20		11:45-12:30	Теория	2	Срезы списков	Каб.№8	Опрос
62	24		11:45-12:30	Теория	2	Списки: примеры решения задач	Каб.№8	Решение задачи
63	27		11:45-12:30	Теория	2	Практическая работа. Решение задач со списками	Каб.№8	Решение задачи
64	Май	1	11:45-12:30	Теория	2	Матрицы	Каб.№8	Опрос
65		4	11:45-12:30	Теория	2	Тест. Списки	Каб.№8	Тест
66		8	11:45-12:30	Теория	2	Кортежи	Каб.№8	Опрос
67		11	11:45-12:30	Теория	2	Введение в словари	Каб.№8	Опрос
68		15	11:45-12:30	Теория	2	Множества в языке Python	Каб.№8	Опрос
69		18	11:45-12:30	Теория	2	Стиль программирования	Каб.№8	Опрос
70		22	11:45-12:30	Теория	2	Отладка программ	Каб.№8	Опрос

71		25	11:45-12:30	Теория	2	Зачет по курсу «Программирование на языке Python»	Каб.№8	Решение задачи, тест
72		29	11:45-12:30	Теория	2	Что дальше?	Каб.№8	Опрос

Календарный учебный график
(для 2-ой группы по программе «Программирование на Python» 1 год обучения)

№ п/п	Месяц	Чи сл о	Время проведения занятия	Форма занятия	Кол- во часо в	Тема занятия	Место проведени я	Форма контрол я
1	Сентябрь	13	18:35-19:20	Введение	2	Инструктаж по технике безопасности в компьютерном классе	Каб.№8	Опрос
2		16	18:35-19:20	Теория	2	Общие сведения о языке	Каб.№8	Опрос
3		20	18:35-19:20	Теория	2	Режимы работы	Каб.№8	Опрос
4		23	18:35-19:20	Практика	2	Знакомство с языком Python	Каб.№8	Решение задачи
5		27	18:35-19:20	Теория	2	Переменные	Каб.№8	Решение задачи
6		30	18:35-19:20	Теория	2	Выражения	Каб.№8	Тест
8	Октябрь	4	18:35-19:20	Теория	2	Ввод и вывод	Каб.№8	Тест
9		7	18:35-19:20	Теория	2	Выражения и операции.	Каб.№8	Решение задачи
10		11	18:35-19:20	Теория	2	Логические выражения и операторы.	Каб.№8	Решение задачи
11		14	18:35-19:20	Теория	2	Условный оператор	Каб.№8	Тест
12		18	18:35-19:20	Теория	2	Условный оператор Решение задачи.	Каб.№8	Решение задачи
13		21	18:35-19:20	Теория	2	Множественное ветвление	Каб.№8	Решение задачи
14		25	18:35-19:20	Теория	2	Множественное ветвление тест.	Каб.№8	Тест
15		28	18:35-19:20	Теория	2	Реализация ветвления в языке Python.	Каб.№8	Решение задачи
16	Ноябрь	1	18:35-19:20	Теория	2	Реализация ветвления в языке Python на практике	Каб.№8	Решение задач
17		4	18:35-19:20	Теория	2	"Составление программ с ветвлением"	Каб.№8	Решение задачи
18		8	18:35-19:20	Теория	2	Зачетная работа. "Составление программ с ветвлением"	Каб.№8	Тест
19		11	18:35-19:20	Теория	2	"Условные операторы".	Каб.№8	Тест
20		15	18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи

21		18	18:35-19:20	Теория	2		Каб.№8	Решение задачи
22		22	18:35-19:20	Теория	2	Оператор цикла for	Каб.№8	Тест
23		25	18:35-19:20	Теория	2	Оператор цикла for закрепление темы	Каб.№8	Решение задачи
24		29	18:35-19:20	Теория	2	Вложенные циклы	Каб.№8	Решение задач
25	Декабрь	2	18:35-19:20	Теория	2	Вложенные циклы закрепление темы	Каб.№8	Опрос
26		6	18:35-19:20	Теория	2	Случайные числа	Каб.№8	Опрос
27		9	18:35-19:20	Теория	2	Случайные числа тест	Каб.№8	Опрос
28		13	18:35-19:20	Теория	2	Самостоятельная работа. "Составление программ с циклом"	Каб.№8	Решение задачи
29		16	18:35-19:20	Теория	2	Циклы	Каб.№8	Опрос
30		20	18:35-19:20	Теория	2	Циклы решение задач.	Каб.№8	
31		23	18:35-19:20	Теория	2	Примеры решения задач с циклом	Каб.№8	Решение задачи
32		27	18:35-19:20	Теория	2	Творческая работа. "Циклы"	Каб.№8	Решение задачи
33		30	18:35-19:20	Теория	2	Цикл тест	Каб.№8	Тест
34		Январь	10	18:35-19:20	Теория	2	Закрепление темы «Цикл»	Каб.№8
35	13		18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
36	17		18:35-19:20	Теория	2	Цикл	Каб.№8	Решение задачи
37	20		18:35-19:20	Теория	2	Условия	Каб.№8	Опрос
38	24		18:35-19:20	Теория	2	Создание функций	Каб.№8	Опрос
39	27		18:35-19:20	Теория	2	Создание функции закрепление информации	Каб.№8	Опрос
40	31		18:35-19:20	Теория	2	Создание функций тестовое задание	Каб.№8	Тест
41	Февраль		3	18:35-19:20	Теория	2	Практическая работа Создание функций	Каб.№8
42		7	18:35-19:20	Теория	2	Локальные переменные	Каб.№8	опрос
43		10	18:35-19:20	Теория	2	Локальные переменные Тестовые задания	Каб.№8	Тест
44		14	18:35-19:20	Теория	2	Примеры решения задач с использованием функций	Каб.№8	Тест
45		17	18:35-19:20	Теория	2	Практическая работа Решение	Каб.№8	Решение задачи

						задач с использованием функций		
46		21	18:35-19:20	Теория	2	Самостоятельная работа по теме "Функции"	Каб.№8	Тест
47		24	18:35-19:20	Теория	2	Рекурсивные функции	Каб.№8	Тест
48		28	18:35-19:20	Теория	2	Рекурсивные функции Решение задачи	Каб.№8	Тест
49	Март	3	18:35-19:20	Теория	2	Практическая работа Рекурсивные функции	Каб.№8	Решение задачи
50		7	18:35-19:20	Теория	2	Тест Функции	Каб.№8	Тест
51		10	18:35-19:20	Теория	2	Строки	Каб.№8	Опрос
52		14	18:35-19:20	Теория	2	Строки продолжение	Каб.№8	Опрос
53		17	18:35-19:20	Теория	2	Практическая работа: Строки	Каб.№8	Решение задачи
54		21	18:35-19:20	Теория	2	Строки Решение задач	Каб.№8	Решение задачи
55		24	18:35-19:20	Теория	2	Срезы строк	Каб.№8	Опрос
56		28	18:35-19:20	Теория	2	Срезы строк Решение задач	Каб.№8	Решение задачи
57		31	18:35-19:20	Теория	2	Примеры решения задач со строками	Каб.№8	Решение задачи
58		Апрель	4	18:35-19:20	Теория	2	Практическая работа: Решение задач со строками.	Каб.№8
59	7		18:35-19:20	Теория	2	Списки	Каб.№8	Опрос
60	11		18:35-19:20	Теория	2	Практическая работа: Списки	Каб.№8	Решение задачи
61	14		18:35-19:20	Теория	2	Срезы списков	Каб.№8	Опрос
62	18		18:35-19:20	Теория	2	Списки: примеры решения задач	Каб.№8	Решение задачи
63	21		18:35-19:20	Теория	2	Практическая работа. Решение задач со списками	Каб.№8	Решение задачи
64	25		18:35-19:20	Теория	2	Матрицы	Каб.№8	Опрос
65	28		18:35-19:20	Теория	2	Тест. Списки	Каб.№8	Тест
66	Май	2	18:35-19:20	Теория	2	Кортежи	Каб.№8	Опрос
67		5	18:35-19:20	Теория	2	Введение в словари	Каб.№8	Опрос
68		9	18:35-19:20	Теория	2	Множества в языке Python	Каб.№8	Опрос
69		12	18:35-19:20	Теория	2	Стиль программирования	Каб.№8	Опрос
70		16	18:35-19:20	Теория	2	Отладка программ	Каб.№8	Опрос

71		19	18:35-19:20	Теория	2	Зачет по курсу «Программирование на языке Python»	Каб.№8	Решение задачи, тест
72		23	18:35-19:20	Теория	2	Что дальше?	Каб.№8	Опрос

Календарный учебный график
(для 3-ой группы по программе «Программирование на Python» 1 год обучения)

№ п/п	Месяц	Чи сл о	Время проведения занятия	Форма занятия	Кол- во часо в	Тема занятия	Место проведени я	Форма контрол я	
1	Сентябрь	10	18:35-19:20	Введение	2	Инструктаж по технике безопасности в компьютерном классе	Каб.№8	Опрос	
2		14	18:35-19:20	Теория	2	Общие сведения о языке	Каб.№8	Опрос	
3		17	18:35-19:20	Теория	2	Режимы работы	Каб.№8	Опрос	
4		21	18:35-19:20	Практика	2	Знакомство с языком Python	Каб.№8	Решение задачи	
5		24	18:35-19:20	Теория	2	Переменные	Каб.№8	Решение задачи	
6		28	18:35-19:20	Теория	2	Выражения	Каб.№8	Тест	
8		Октябрь	1	18:35-19:20	Теория	2	Ввод и вывод	Каб.№8	Тест
9	5		18:35-19:20	Теория	2	Выражения и операции.	Каб.№8	Решение задачи	
10	8		18:35-19:20	Теория	2	Логические выражения и операторы.	Каб.№8	Решение задачи	
11	12		18:35-19:20	Теория	2	Условный оператор	Каб.№8	Тест	
12	15		18:35-19:20	Теория	2	Условный оператор Решение задачи.	Каб.№8	Решение задачи	
13	19		18:35-19:20	Теория	2	Множественное ветвление	Каб.№8	Решение задачи	
14	22		18:35-19:20	Теория	2	Множественное ветвление тест.	Каб.№8	Тест	
15	26		18:35-19:20	Теория	2	Реализация ветвления в языке Python.	Каб.№8	Решение задачи	
16	29		18:35-19:20	Теория	2	Реализация ветвления в языке Python на практике	Каб.№8	Решение задач	
17	Ноябрь		2	18:35-19:20	Теория	2	"Составление программ с ветвлением"	Каб.№8	Решение задачи
18			9	18:35-19:20	Теория	2	Зачетная работа. "Составление программ с ветвлением"	Каб.№8	Тест
19		12		Теория	2	"Условные	Каб.№8	Тест	

			18:35-19:20			операторы".			
20		16	18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи	
21		19	18:35-19:20	Теория	2		Каб.№8	Решение задачи	
22		23	18:35-19:20	Теория	2	Оператор цикла for	Каб.№8	Тест	
23		26	18:35-19:20	Теория	2	Оператор цикла for закрепление темы	Каб.№8	Решение задачи	
24		30	18:35-19:20	Теория	2	Вложенные циклы	Каб.№8	Решение задач	
25	Декабрь	3	18:35-19:20	Теория	2	Вложенные циклы закрепление темы	Каб.№8	Опрос	
26		7	18:35-19:20	Теория	2	Случайные числа	Каб.№8	Опрос	
27		10	18:35-19:20	Теория	2	Случайные числа тест	Каб.№8	Опрос	
28		14	18:35-19:20	Теория	2	Самостоятельная работа. "Составление программ с циклом"	Каб.№8	Решение задачи	
29		17	18:35-19:20	Теория	2	Циклы	Каб.№8	Опрос	
30		21	18:35-19:20	Теория	2	Циклы решение задач.	Каб.№8		
31		24	18:35-19:20	Теория	2	Примеры решения задач с циклом	Каб.№8	Решение задачи	
32		28	18:35-19:20	Теория	2	Творческая работа. "Циклы"	Каб.№8	Решение задачи	
33		31	18:35-19:20	Теория	2	Цикл тест	Каб.№8	Тест	
34		Январь	11	18:35-19:20	Теория	2	Закрепление темы «Цикл»	Каб.№8	Опрос
35			14	18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
36			18	18:35-19:20	Теория	2	Цикл	Каб.№8	Решение задачи
37			21	18:35-19:20	Теория	2	Условия	Каб.№8	Опрос
38	25		18:35-19:20	Теория	2	Создание функций	Каб.№8	Опрос	
39	28		18:35-19:20	Теория	2	Создание функции закрепление информации	Каб.№8	Опрос	
40	Февраль		1	18:35-19:20	Теория	2	Создание функций тестовое задание	Каб.№8	Тест
41		4	18:35-19:20	Теория	2	Практическая работа Создание функций	Каб.№8	Решение задачи	
42		8	18:35-19:20	Теория	2	Локальные переменные	Каб.№8	опрос	
43		11	18:35-19:20	Теория	2	Локальные переменные Тестовые задания	Каб.№8	Тест	
44		15	18:35-19:20	Теория	2	Примеры решения задач с использованием	Каб.№8	Тест	

						функций			
45		18	18:35-19:20	Теория	2	Практическая работа Решение задач с использованием функций	Каб.№8	Решение задачи	
46		22	18:35-19:20	Теория	2	Самостоятельная работа по теме "Функции"	Каб.№8	Тест	
47		25	18:35-19:20	Теория	2	Рекурсивные функции	Каб.№8	Тест	
48	Март	1	18:35-19:20	Теория	2	Рекурсивные функции Решение задачи	Каб.№8	Тест	
49		4	18:35-19:20	Теория	2	Практическая работа Рекурсивные функции	Каб.№8	Решение задачи	
50		11	18:35-19:20	Теория	2	Тест Функции	Каб.№8	Тест	
51		15	18:35-19:20	Теория	2	Строки	Каб.№8	Опрос	
52		18	18:35-19:20	Теория	2	Строки продолжение	Каб.№8	Опрос	
53		22	18:35-19:20	Теория	2	Практическая работа: Строки	Каб.№8	Решение задачи	
54		25	18:35-19:20	Теория	2	Строки Решение задач	Каб.№8	Решение задачи	
55		29	18:35-19:20	Теория	2	Срезы строк	Каб.№8	Опрос	
56		Апрель	1	18:35-19:20	Теория	2	Срезы строк Решение задач	Каб.№8	Решение задачи
57			5	18:35-19:20	Теория	2	Примеры решения задач со строками	Каб.№8	Решение задачи
58	8		18:35-19:20	Теория	2	Практическая работа: Решение задач со строками.	Каб.№8	Решение задачи	
59	12		18:35-19:20	Теория	2	Списки	Каб.№8	Опрос	
60	15		18:35-19:20	Теория	2	Практическая работа: Списки	Каб.№8	Решение задачи	
61	19		18:35-19:20	Теория	2	Срезы списков	Каб.№8	Опрос	
62	22		18:35-19:20	Теория	2	Списки: примеры решения задач	Каб.№8	Решение задачи	
63	26		18:35-19:20	Теория	2	Практическая работа. Решение задач со списками	Каб.№8	Решение задачи	
64	29		18:35-19:20	Теория	2	Матрицы	Каб.№8	Опрос	
65	Май		3	18:35-19:20	Теория	2	Тест. Списки	Каб.№8	Тест
66		6	18:35-19:20	Теория	2	Кортежи	Каб.№8	Опрос	
67		10	18:35-19:20	Теория	2	Введение в словари	Каб.№8	Опрос	
68		13		Теория	2	Множества в	Каб.№8	Опрос	

		18:35-19:20			языке Python		
69		17 18:35-19:20	Теория	2	Стиль программирования	Каб.№8	Опрос
70		20 18:35-19:20	Теория	2	Отладка программ	Каб.№8	Опрос
71		24 18:35-19:20	Теория	2	Зачет по курсу «Программирование на языке Python»	Каб.№8	Решение задачи, тест
72		27 18:35-19:20	Теория	2	Что дальше?	Каб.№8	Опрос

Календарный учебный график

(для 4-ой группы по программе «Программирование на Python» 1 год обучения)

№ п/п	Месяц	Число	Время проведения занятия	Форма занятия	Кол-во часов	Тема занятия	Место проведения	Форма контроля
1	Сентябрь	12	18:35-19:20	Введение	2	Инструктаж по технике безопасности в компьютерном классе	Каб.№8	Опрос
2		15	18:35-19:20	Теория	2	Общие сведения о языке	Каб.№8	Опрос
3		19	18:35-19:20	Теория	2	Режимы работы	Каб.№8	Опрос
4		22	18:35-19:20	Практика	2	Знакомство с языком Python	Каб.№8	Решение задачи
5		26	18:35-19:20	Теория	2	Переменные	Каб.№8	Решение задачи
6		29	18:35-19:20	Теория	2	Выражения	Каб.№8	Тест
8	Октябрь	3	18:35-19:20	Теория	2	Ввод и вывод	Каб.№8	Тест
9		6	18:35-19:20	Теория	2	Выражения и операции.	Каб.№8	Решение задачи
10		10	18:35-19:20	Теория	2	Логические выражения и операторы.	Каб.№8	Решение задачи
11		13	18:35-19:20	Теория	2	Условный оператор	Каб.№8	Тест
12		17	18:35-19:20	Теория	2	Условный оператор Решение задачи.	Каб.№8	Решение задачи
13		20	18:35-19:20	Теория	2	Множественное ветвление	Каб.№8	Решение задачи
14		24	18:35-19:20	Теория	2	Множественное ветвление тест.	Каб.№8	Тест
15		27	18:35-19:20	Теория	2	Реализация ветвления в языке Python.	Каб.№8	Решение задачи
16		Ноябрь	31	18:35-19:20	Теория	2	Реализация ветвления в языке Python на практике	Каб.№8
17	3		18:35-19:20	Теория	2	"Составление программ с ветвлением"	Каб.№8	Решение задачи

18		7	18:35-19:20	Теория	2	Зачетная работа. "Составление программ с ветвлением"	Каб.№8	Тест
19		10	18:35-19:20	Теория	2	"Условные операторы".	Каб.№8	Тест
20		14	18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
21		17	18:35-19:20	Теория	2		Каб.№8	Решение задачи
22		21	18:35-19:20	Теория	2	Оператор цикла for	Каб.№8	Тест
23		24	18:35-19:20	Теория	2	Оператор цикла for закрепление темы	Каб.№8	Решение задачи
24		28	18:35-19:20	Теория	2	Вложенные циклы	Каб.№8	Решение задач
25	Декабрь	1	18:35-19:20	Теория	2	Вложенные циклы закрепление темы	Каб.№8	Опрос
26		5	18:35-19:20	Теория	2	Случайные числа	Каб.№8	Опрос
27		8	18:35-19:20	Теория	2	Случайные числа тест	Каб.№8	Опрос
28		12	18:35-19:20	Теория	2	Самостоятельная работа. "Составление программ с циклом"	Каб.№8	Решение задачи
29		15	18:35-19:20	Теория	2	Циклы	Каб.№8	Опрос
30		19	18:35-19:20	Теория	2	Циклы решение задач.	Каб.№8	
31		22	18:35-19:20	Теория	2	Примеры решения задач с циклом	Каб.№8	Решение задачи
32		26	18:35-19:20	Теория	2	Творческая работа. "Циклы"	Каб.№8	Решение задачи
33		29	18:35-19:20	Теория	2	Цикл тест	Каб.№8	Тест
34		Январь	9	18:35-19:20	Теория	2	Закрепление темы «Цикл»	Каб.№8
35	12		18:35-19:20	Теория	2	Оператор цикла с условием	Каб.№8	Решение задачи
36	16		18:35-19:20	Теория	2	Цикл	Каб.№8	Решение задачи
37	19		18:35-19:20	Теория	2	Условия	Каб.№8	Опрос
38	23		18:35-19:20	Теория	2	Создание функций	Каб.№8	Опрос
39	28		18:35-19:20	Теория	2	Создание функции закрепление информации	Каб.№8	Опрос
40	30		18:35-19:20	Теория	2	Создание функций тестовое задание	Каб.№8	Тест
41	Февраль	2	18:35-19:20	Теория	2	Практическая работа Создание функций	Каб.№8	Решение задачи
42		6	18:35-19:20	Теория	2	Локальные переменные	Каб.№8	опрос
43		9		Теория	2	Локальные	Каб.№8	Тест

			18:35-19:20			переменные Тестовые задания		
44		13	18:35-19:20	Теория	2	Примеры решения задач с использованием функций	Каб.№8	Тест
45		16	18:35-19:20	Теория	2	Практическая работа Решение задач с использованием функций	Каб.№8	Решение задачи
46		27	18:35-19:20	Теория	2	Самостоятельная работа по теме "Функции"	Каб.№8	Тест
47	Март	2	18:35-19:20	Теория	2	Рекурсивные функции	Каб.№8	Тест
48		6	18:35-19:20	Теория	2	Рекурсивные функции Решение задачи	Каб.№8	Тест
49		9	18:35-19:20	Теория	2	Практическая работа Рекурсивные функции	Каб.№8	Решение задачи
50		13	18:35-19:20	Теория	2	Тест Функции	Каб.№8	Тест
51		16	18:35-19:20	Теория	2	Строки	Каб.№8	Опрос
52		20	18:35-19:20	Теория	2	Строки продолжение	Каб.№8	Опрос
53		23	18:35-19:20	Теория	2	Практическая работа: Строки	Каб.№8	Решение задачи
54		27	18:35-19:20	Теория	2	Строки Решение задач	Каб.№8	Решение задачи
55		30	18:35-19:20	Теория	2	Срезы строк	Каб.№8	Опрос
56		Апрель	3	18:35-19:20	Теория	2	Срезы строк Решение задач	Каб.№8
57	6		18:35-19:20	Теория	2	Примеры решения задач со строками	Каб.№8	Решение задачи
58	10		18:35-19:20	Теория	2	Практическая работа: Решение задач со строками.	Каб.№8	Решение задачи
59	13		18:35-19:20	Теория	2	Списки	Каб.№8	Опрос
60	17		18:35-19:20	Теория	2	Практическая работа: Списки	Каб.№8	Решение задачи
61	20		18:35-19:20	Теория	2	Срезы списков	Каб.№8	Опрос
62	24		18:35-19:20	Теория	2	Списки: примеры решения задач	Каб.№8	Решение задачи
63	27		18:35-19:20	Теория	2	Практическая работа. Решение задач со списками	Каб.№8	Решение задачи
64	Май	1	18:35-19:20	Теория	2	Матрицы	Каб.№8	Опрос
65		4	18:35-19:20	Теория	2	Тест. Списки	Каб.№8	Тест

66		8	18:35-19:20	Теория	2	Кортежи	Каб.№8	Опрос
67		11	18:35-19:20	Теория	2	Введение в словари	Каб.№8	Опрос
68		15	18:35-19:20	Теория	2	Множества в языке Python	Каб.№8	Опрос
69		18	18:35-19:20	Теория	2	Стиль программирования	Каб.№8	Опрос
70		22	18:35-19:20	Теория	2	Отладка программ	Каб.№8	Опрос
71		24	18:35-19:20	Теория	2	Зачет по курсу «Программирование на языке Python»	Каб.№8	Решение задачи, тест
72		28	18:35-19:20	Теория	2	Что дальше?	Каб.№8	Опрос

2.2. Условия реализации программы

Материально-техническое обеспечение:

- столы для компьютера;
- компьютерные стулья;
- шкафы для дидактических материалов, пособий;
- специальная и научно-популярная литература для педагога и учащихся;
- канцтовары;

Информационное обеспечение:

- персональный компьютер (на каждого участника);
- мультимедийный проектор;
- видеоматериалы разной тематики по программе;
- оргтехника;
- выход в сеть Internet;

Аппаратное обеспечение:

- Процессор не ниже Core2 Duo;
- Объем оперативной памяти не ниже 4 Гб DDR3;
- Дисковое пространство на менее 128 Гб;
- Монитор диагональю на мене 19”;

Программное обеспечение:

- Операционная система Windows 7 Профессиональная или выше;
- Интерпретатор Python версии 3.7 и выше;
- IDE JatBrains PyCharm;
- Foxit Reader или другой просмотрщик PDF файлов;
- WinRAR;
- Пакет офисных программ;
- Adobe Photoshop или другой растровый графический редактор;
- Любой браузер для интернет серфинга.

Кадровое обеспечение

Реализовывать программу может педагог дополнительного образования, обладающий достаточными знаниями в области педагогики, психологии и методологии, знающий особенности технологии обучения основам программирования на языке Python.

2.3. Формы аттестации

Входящий контроль осуществляется при комплектовании группы в начале учебного года.

Цель – определить исходный уровень знаний учащихся, определить формы и методы работы с учащимися.

Форма контроля: тестирование.

Текущий контроль осуществляется после изучения отдельных тем, раздела программы. В практической деятельности результативность оценивается качеством выполнения практических работ, поиску и отбору необходимого материала, умению работать с различными источниками информации. Анализируются положительные и отрицательные стороны работы, корректируются недостатки. Контроль знаний осуществляется с помощью заданий педагога (решение практических задач средствами языка программирования); взаимоконтроля, самоконтроля и др. Они активизируют, стимулируют работу учащихся, позволяют более полно проявлять полученные знания, умения, навыки.

Промежуточный контроль осуществляется в конце I полугодия учебного года.

Форма контроля: тест, решение практических задач средствами языка программирования.

Итоговый контроль осуществляется в конце учебного года.

Форма контроля: защита творческого проекта.

Отслеживание личностного развития учащихся осуществляется методом наблюдения, анкетирования. По итогам первого полугодия и по итогам года заполняется «Диагностическая карта», в которой проставляется уровень усвоения программы каждым учащимся объединения.

2.4. Оценочные материалы

Входящий контроль: *Тестирование (Приложение 2)*

Промежуточный контроль: *Тестирование, решение задач (Приложение 3)*

Итоговый контроль: *учащимся предлагается самостоятельно выбрать тему творческого проекта и на основании темы разработать программу, пояснительную записку, презентацию.*

2.5. Методические материалы

Форма организации учебного занятия

С целью достижения качественных результатов учебный процесс оснащен современными техническими средствами. Процесс обучения проходит в компьютерном классе. Объясняемый материал демонстрируется на проекторе, все действия производятся преподавателем непосредственно. При обучении используется метод «Делай как я». К каждой теме занятия разработаны презентации теоретического материала. С помощью мультимедийных элементов занятие визуализируется, вызывая положительные эмоции у обучающихся и создавая условия для успешной деятельности каждого ребенка.

Формы работы:

- Занятия

Программа «Программирование на Python» рассчитана на 1 учебный год и построена на преемственности занятий. Знания, полученные на предыдущих занятиях, обучающиеся будут применять на следующих. Программа разработана с учётом возрастных особенностей детей 14-18 лет, основное время отводится на выполнение практических работ под руководством педагога. Самостоятельные действия обучающихся контролируются преподавателем напрямую, и при необходимости вносятся поправки в работу, но самостоятельная работа ученика должна преобладать. Все созданные файлы обязательно сохраняются в процессе работы или в конце урока.

- Урок-игра (игровая деятельность) Использование игрового метода на уроках призвано способствовать созданию благоприятной психологической атмосферы общения.

Игра способствует развитию у детей произвольного внимания, повышает мотивацию к деятельности.

- Урок-проект (проектная деятельность) Метод проектов направлен на то, чтобы развить активное самостоятельное мышление ребенка и научить его не просто запоминать и воспроизводить знания, которые дает ему педагог, а уметь применять их на практике. Проектная методика отличается коллективным характером деятельности, которая при этом является творческой и ориентированной на личность обучающегося. Она предполагает высокий уровень индивидуальной и коллективной ответственности за выполнение каждого задания по разработке проекта. Совместная работа группы учащихся над проектом неотделима от коммуникативного взаимодействия обучающихся. Проект является одной из форм организации исследовательской познавательной деятельности, в которой учащиеся занимают активную позицию. При подборе темы проекта педагог должен ориентироваться на интересы и потребности учащихся, их возможности и личную значимость предстоящей работы, практическую значимость результата работы над проектом. Выполненный проект может быть представлен в самых разных формах. Разнообразны и формы презентации проекта: доклад, дискуссия, демонстрация готового векторного рисунка или обработанной фотографии и др. Главным результатом работы над проектом будут актуализация имеющихся и приобретение новых знаний, навыков и умений и их творческое применение в новых условиях. Работа над проектом требует от учащихся самостоятельной деятельности, координации действий, активного исследовательского, исполнительского и коммуникативного взаимодействия. Роль преподавателя заключается в подготовке учащихся к работе над проектом, выборе темы, в оказании помощи учащимся при планировании работы, в текущем контроле и консультировании учащихся по ходу выполнения проекта.

- Открытое занятие, презентация.

Педагогические технологии и приемы

Технология группового обучения, технология игровой деятельности, технология развивающего обучения, технология проектной деятельности, технология коллективной творческой деятельности, технология сотрудничества.

Для привлечения внимания ребенка необходимо не только разнообразие деятельности во время учебного процесса, но также и личностно-ориентированный подход к каждому ребенку. Именно такой подход позволит ребенку раскрыть и развить свои личностные качества.

3. Список литературы

Литература для педагога

1. Бизли, Дэвид М. Python. Подробный справочник. – М.–СПб.: Символ-Плюс, 2010.
2. Лутц, Марк Python. Справочник. – М.: Вильямс, 2015.
3. Официальный сайт программы [Электронный ресурс] – Режим доступа: <https://docs.python.org/>, свободный.
4. Сайт, среда разработки для языка Python. [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu>, свободный.

Литература для родителей и учащихся

1. Сайт / справочные материалы [Электронный ресурс] – Режим доступа: <https://metanit.com/python/>, свободный.
2. Сайт / интерактивный сборник задач для практики программирования [Электронный ресурс] – Режим доступа: <http://pythontutor.ru/>, свободный.
3. Сайт / Адаптивный тренажер Python [Электронный ресурс] – Режим доступа: <https://stepik.org/course/431> , свободный.
4. Сайт / среда разработки для языка Python [Электронный ресурс] – Режим доступа: <https://www.jetbrains.com/pycharm/?fromMenu> , свободный.

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

Входящий контроль

Низкий уровень: 0-70% выполненных заданий;

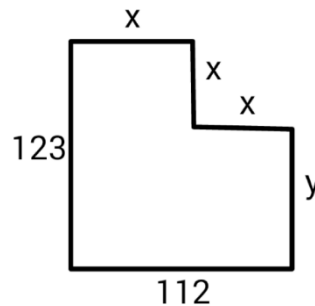
Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

Для учащихся – 14-17 лет

Фамилия, Имя	
---------------------	--

1. Из прямоугольника вырезали квадрат со стороной x , получилась фигура как на рисунке. Чему равна сумма цифр y ?



2. В алфавите племени мумба-юмба 32 буквы. Любое слово в языке этого племени состоит из пяти букв и должно одинаково читаться справа налево и слева направо, при этом первые две буквы слова обязательно различаются, а третья совпадает с пятой.

Каково максимальное количество слов в этом языке?

3. Катя наклеила на рулет тонкие поперечные кольца трёх разных цветов. Если разрезать по серым кольцам, получится 25 кусков рулета, если по малиновым — 47 кусков, а если по зеленым — 31 кусок.

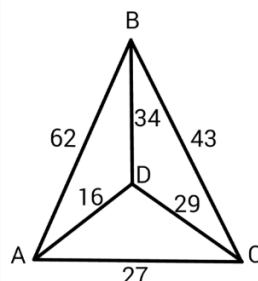
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

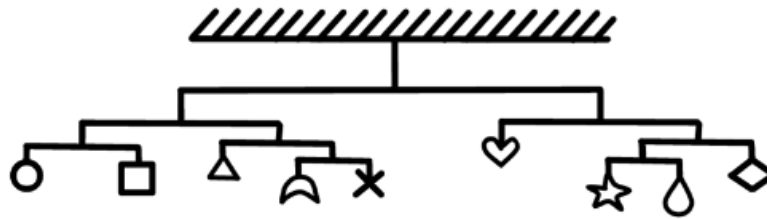
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

4. На рисунке показано расположение городов А, В, С и D и расстояния между ними. Турист выходит из города В и собирается посетить остальные города, побывав в каждом по разу.

Какова наименьшая возможная длина маршрута, если он хочет закончить свой путь в том же городе?



5. Фигурки, общей массой 432 грамма, при помощи невесомых нитей и планок собрали в конструкцию, изображённую на рисунке. Оказалось, что все её части находятся в равновесии. Сколько весит ромб?



6. Люди переезжают в города, за год численность людей удваивается. Если люди заселят весь город за 12 лет, то сколько лет понадобится, чтобы занять лишь половину города?

7. Сравните пары слов. Сколько среди них полностью идентичных?

O/Sanmarco

O/Samnarco

Ф.Wagonerrte

Ф.Wagonertre

A.S.Schmetterling

A.S.Schnetterling

N.V.Murfreesboroque

N.V.Munfreesboroque

P.S.Splendoursec

P.S.Sqlendoursec

Семь человек выясняли, какой сегодня день недели.

Первый сказал: «Послезавтра – воскресенье».

Второй: «Вчера был понедельник».

Третий: «Завтра будет суббота».

Четвертый: «Завтра будет среда».

Пятый: «Вчера был четверг».

Шестой: «Позавчера было воскресенье».

Седьмой: «Позавчера была среда».

Какой сегодня день недели, если трое ошибаются?

8. Вам предложены несколько высказываний и следствие из них (выделено жирным).

Согласны ли Вы с этим следствием?

1. Все клёны — растения.

2. Некоторые растения быстро желтеют.

Значит, некоторые клёны быстро желтеют.

Да

Нет

9. Гусеница прогрызает яблоко диаметром 6 сантиметров насквозь за 16 секунд, вылезая снаружи полностью.

Известно, что середину яблока она начинает грызть уже через 6 секунд после начала пути.

Какова длина гусеницы в сантиметрах?

Для учащихся 14-18 лет

Фамилия, Имя.	
---------------	--

1. Назовите два числа, у которых количество цифр равно количеству букв, составляющих название каждого из этих чисел.

--	--

2. Собака была привязана к десятиметровой веревке, а прошла двести метров. Как ей это удалось?

--

3. Есть дорога, по которой может проехать только одна машина. По дороге едут две машины: одна с горы, другая под гору. Как им разъехаться?

--

4. Как с помощью только одной палочки образовать на столе треугольник?

--

5. В каком месяце болтливая девочка говорит меньше всего?

--

6. Что становится на треть больше, если его поставить вверх ногами?

--

7. Представьте себе, что вы кондуктор. Поезд везет сто вагонов, в каждом вагоне 10 купе, в каждом купе 4 пассажира. Сколько лет кондуктору?

--

8. Юра разрезал огромную пиццу на 10 кусков. Затем он взял один из кусков и разрезал его еще на 10. После этого из имеющихся кусков он выбрал два и разрезал каждый из них на 10.

Вопрос: Сколько кусков пиццы получилось у Юры.

--

9. В компьютерной игре нужно победить монстра. Изначально у Юры было только 9 выстрелов. Но за каждое попадание он получал дополнительно еще 3 выстрела.

Вопрос: Сколько раз Юра попал в цель, если всего он выстрелил 30 раз, израсходовав все выстрелы?

--

Итоговый контроль
Диагностическая карта

№/ п	Имя	Входящая Диагностика	Промежуточ- ная диагностика	Итоговая диагностика
1				
2				
3+				
	Итого в %	Н С В		Н С В

Результативность отслеживается с помощью карты наблюдений, анализа участия детей в совместной продуктивной деятельности, разработки и защите творческих проектов. Усвоение программы возможно по 3-м уровням: низкий (Н), средний (С), высокий (В).

Низкий уровень

Учащиеся должны знать / понимать:

- технику безопасного поведения во время занятий;
- правила поведения в общественных местах,
- понятие программы Python; общую структуру программы;
- основные типы данных;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- основные циклы с условием;
- основные правила записи циклов условием;
- формат записи цикла с параметром;
- понятие функции;
- основные принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- операции со строками;
- способ описания списка;
- способ описания кортежа;
- способ описания словаря;
- основные операции, выполняемые со списками, кортежами и словарями;
- понятие множества;
- способы описания множества;
- операторы работы с множествами.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ.

Учащиеся должны уметь:

- понимать учебную задачу, сохранять ее содержание в процессе ее выполнения под руководством педагога;
- работать в паре, малой группе;
- выполнить установку программы под руководством педагога;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе под руководством педагога;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием под руководством педагога;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи по наводящим вопросам педагога;
- создавать и использовать основные функции;
- описывать и соединять строки;
- находить подстроку в строке с помощью педагога;
- находить количество слов в строке;
- вводить и выводить элементы списка под руководством педагога;
- приводить примеры использования вложенных списков (матриц) по наводящим вопросам педагога;
- описывать множества под руководством педагога;
- определять принадлежность элемента множеству по наводящим вопросам;
- определять вид ошибок и находить ошибки в программе под руководством педагога;
- составлять элементарные алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python под руководством педагога.

Средний уровень

Учащиеся должны знать / понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности;
- основные приемы взаимодействия в группе сверстников;
- понятие программы;
- структуру программы на Python;
- режимы работы с Python.
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- циклы с условием и их виды;
- назначение и особенности использования цикла с параметром;
- формат записи цикла с параметром;
- примеры использования циклов различных типов.
- понятие функции;
- основные способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.

- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- 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.
- что такое стиль программирования;
- правила именования объектов;
- основные рекомендации при написании программ;
- основные шаги работы над проектом, его презентации.

Учащиеся должны уметь:

- уважительно относиться к преподавателям и сверстникам;
- применять некоторые приемы логического (абстрактное) мышления;
- концентрировать внимание на одном или двух объектах;
- понимать причины успеха/неуспеха с помощью анализа педагога;
- выполнить установку программы;
- выполнить простейшую программу в интерактивной среде;
- написать комментарии в программе;
- решать задачи на элементарные действия с числами;
- использовать условный оператор;
- создавать сложные условия с помощью логических операторов;
- определять вид цикла, наиболее удобный для решения поставленной задачи;
- использовать цикл с условием;
- определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
- создавать и использовать функции;
- использовать механизм параметров для передачи значений;
- описывать строки;
- соединять строки;
- находить длину строки;
- вырезать часть строки;
- находить подстроку в строке;
- находить количество слов в строке;
- 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.
- определять вид ошибок и находить ошибки в программе.

- составлять алгоритмы для решения задач;
- реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
- отлаживать и тестировать программы, написанные на языке Python;
- понимать и сохранять цели и задачи учебной деятельности в ходе занятия;
- планировать свою деятельность с помощью взрослого;
- сотрудничать с взрослыми и сверстниками в процессе выполнения учебной задачи;
- понимать причины успеха/неуспеха учебной деятельности под руководством взрослого;
- делать выводы в ходе полученных заданий; выстраивать логические цепи рассуждений под руководством педагога;
- выражать творческие идеи, разработать творческий проект на основе образца;
- конструктивно взаимодействовать в составе группы в ходе работы над проектом.

Высокий уровень

Учащиеся должны знать / понимать:

- основные положения техники безопасности на занятиях, правила поведения в общественных местах, правила дорожной безопасности, правила поведения во время чрезвычайных происшествий;
- приемы конструктивного взаимодействия в группе сверстников;
- понятие программы;
- структуру программы на Python;
- режимы работы с Python.
- общую структуру программы;
- типы данных;
- целые, вещественные типы данных и операции над ними;
- оператор присваивания;
- назначение условного оператора;
- способ записи условного оператора;
- логический тип данных;
- логические операторы or, and, not;
- 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.
- понятие функции;
- способы описания функции;
- принципы структурного программирования;
- понятие локальных переменных подпрограмм;
- понятие формальных и фактических параметров подпрограмм;
- способ передачи параметров.
- назначение строкового типа данных;
- операторы для работы со строками;
- процедуры и функции для работы со строками;
- операции со строками;
- 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.
 - что такое стиль программирования;
 - правила именования объектов;
 - основные рекомендации при написании программ;
 - правила и этапы работы над проектом;
 - приемы успешной презентации проекта.
- Учащиеся должны уметь:*
- выполнить установку программы;
 - выполнить простейшую программу в интерактивной среде;
 - написать комментарии в программе;
 - решать задачи на элементарные действия с числами;
 - использовать условный оператор;
 - создавать сложные условия с помощью логических операторов;
 - определять вид цикла, наиболее удобный для решения поставленной задачи;
 - использовать цикл с условием;
 - определять целесообразность применения и использования цикла с параметром для решения поставленной задачи;
 - создавать и использовать функции;
 - использовать механизм параметров для передачи значений;
 - описывать строки;
 - соединять строки;
 - находить длину строки;
 - вырезать часть строки;
 - находить подстроку в строке;
 - находить количество слов в строке;
 - 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.
 - определять вид ошибок и находить ошибки в программе.
 - составлять алгоритмы для решения задач;
 - реализовывать алгоритмы на компьютере в виде программ, написанных на языке Python;
 - отлаживать и тестировать программы, написанные на языке Python;
 - понимать учебную задачу, анализировать достижение результата;
 - делать выводы в ходе полученных заданий; самостоятельно выстраивать логические цепи рассуждений;
 - понимать причины успеха/неуспеха, конструктивно принимать причины неуспеха, исправлять ошибки с минимальным участием педагога;
 - планировать свою деятельность, находить оригинальные способы выполнения поставленной творческой задачи;
 - создать проектировочную команду и организовать ее деятельность;
 - разработать творческий проект по предложенной или самостоятельно выбранной теме в составе творческой группы;
 - самостоятельно подготовить презентацию и защитить проект.

Вопросы для промежуточного контроля по усвоению материала
(Промежуточный контроль проводится в последнюю неделю декабря)

Составьте выражение для вычисления в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления:

11111·11111111

— **произведение чисел 11111 (5 единиц) и 1111111 (7 единиц)**

Запишите число **1.2345e3** в виде **десятичной** дроби.

Составьте и запишите выражение для вычисления:

2014.0¹⁴(Возвестив 14 степень)

Обратите внимание на запись числа: это вещественное число.

Приведите к целому типу число 2.99

Расставьте скобки в выражении

a and b or not a and not b

в соответствии с порядком вычисления выражения (приоритетом операций).
Всего потребуется 5 пар скобок (внешние скобки входят в их число).

Найдите результат выражения для заданных значений *a* и *b*

Учитывайте регистр символов при ответе.

a = True

b = False

a and b or not a and not b

Отметьте выражения, значения которых равны True:

- "239" < "30" and 239 < 30
- "239" < "30" and 239 > 30
- "239" > "30" and 239 < 30
- "239" > "30" and 239 > 30

Укажите результат выражения:

"123" + "42"

Какое значение будет у переменной *i* после выполнения фрагмента программы?

```
i = 0
```

```
while i <= 10:
```

```
    i = i + 1
```

```
    if i > 7:
```

```
        i = i + 2
```

Сколько итераций цикла будет выполнено в этом фрагменте программы?

```
i = 0
```

```
while i <= 10:
```

```
    i = i + 1
```

```
    if i > 7:
```

```
        i = i + 2
```

Сколько всего знаков * будет выведено после исполнения фрагмента программы:

```
i = 0  
while i < 5:  
print('*')  
    if i % 2 == 0:  
        print('*')  
    if i > 2:  
        print('***')  
    i = i + 1
```

Определите, какое значение будет иметь переменная i после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i < 10:  
    i = i + 1  
    s = s + i  
    if s > 15:  
        break  
    i = i + 1
```

Определите, какое значение будет иметь переменная i после выполнения следующего фрагмента программы:

```
i = 0  
s = 0  
while i < 10:  
    i = i + 1  
    s = s + i  
    if s > 15:  
        continue  
    i = i + 1
```

Задачи для промежуточного контроля по усвоению материала

Напишите простой калькулятор, который считывает с пользовательского ввода три строки: первое число, второе число и операцию, после чего применяет операцию к введённым числам ("первое число" "операция" "второе число") и выводит результат на экран.

Поддерживаемые операции: +, -, /, *, mod, pow, div, где
mod — это взятие остатка от деления,
pow — возведение в степень,
div — целочисленное деление.

Если выполняется деление и второе число равно 0, необходимо выводить строку "Деление на 0!".

Обратите внимание, что на вход программе приходят вещественные числа.

Жители страны Малевии часто экспериментируют с планировкой комнат. Комнаты бывают треугольные, прямоугольные и круглые. Чтобы быстро вычислять жилплощадь, требуется написать программу, на вход которой подаётся тип фигуры комнаты и соответствующие параметры, которая бы выводила площадь получившейся комнаты.

Для числа π в стране Малевии используют значение 3.14.

Формат ввода, который используют Малевийцы:

Треугольник

a

b

c

где a, b и c — длины сторон треугольника

прямоугольник

a

b

где a и b — длины сторон прямоугольника

круг

r

где r — радиус окружности

Напишите программу, которая получает на вход три целых числа, по одному числу в строке, и выводит на консоль в три строки сначала максимальное, потом минимальное, после чего оставшееся число.

На ввод могут подаваться и повторяющиеся числа.

Паша очень любит кататься на общественном транспорте, а получая билет, сразу проверяет, счастливый ли ему попался. Билет считается счастливым, если сумма первых трех цифр совпадает с суммой последних трех цифр номера билета.

Однако Паша очень плохо считает в уме, поэтому попросил вас написать программу, которая проверит равенство сумм и выведет "Счастливый", если суммы совпадают, и "Обычный", если суммы различны.

На вход программе подаётся строка из шести цифр. **(Пример:123321)**

Выводить нужно только слово "Счастливый" или "Обычный", с большой буквы.

Напишите программу, которая считывает с консоли числа (по одному в строке) до тех пор, пока сумма введённых чисел не будет равна 0 и **сразу после этого** выводит сумму квадратов всех считанных чисел.

Гарантируется, что в какой-то момент сумма введённых чисел окажется равной 0, **после этого считывание продолжать не нужно.**

В примере мы считываем числа 1, -3, 5, -6, -10, 13; в этот момент замечаем, что сумма этих чисел равна нулю и выводим сумму их квадратов, не обращая внимания на то, что остались ещё не прочитанные значения.

Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 5 5 5 ... (число повторяется столько раз, чему равно). На вход программе передаётся неотрицательное целое число n — столько элементов последовательности должна отобразить программа. На выходе ожидается последовательность чисел, записанных через пробел в одну строку.

Например, если $n = 7$, то программа должна вывести 1 2 2 3 3 3 4.

Выведите таблицу размером $n \times n$, заполненную числами от **1 до n^2** по спирали, выходящей из левого верхнего угла и закрученной по часовой стрелке, как показано в примере (**здесь $n=5$**)