

Департамент Смоленской области по образованию и науке  
муниципальное бюджетное общеобразовательное учреждение  
«Остерская средняя школа»

ПРИНЯТА  
на заседании педагогического  
совета от 31.08.2022  
Протокол № 14

УТВЕРЖДАЮ  
Директор МБОУ «Остерская  
средняя школа»  
Т.И. Терехова  
01.09.2022г. Приказ №101



Дополнительная общеобразовательная общеразвивающая  
программа технической направленности  
«Введение в язык программирования Python»

Возраст обучающихся: 11-15 лет  
Срок реализации: 1 год

Автор-составитель:  
педагог дополнительного образования,  
Корнеев Сергей Владиславович

С. Остер 2022

## Пояснительная записка

Дополнительная общеобразовательная общеразвивающая программа «Введение в язык программирования Python» разработана в соответствии с нормативно – правовыми документами:

- Законом «Об образовании в Российской Федерации» от 29 декабря 2012 г. № 273-ФЗ;

- Об утверждении Порядка организации и осуществления образовательной деятельности по дополнительным общеобразовательным программам (Приказ Минпрос РФ от 9 ноября 2018 г. № 196);

- СанПиН 2.4. 3648-20 «Санитарно-эпидемиологические требования к организациям воспитания и обучения, отдыха и оздоровления детей и молодежи» (Постановление Главного государственного санитарного врача РФ от 28.09. 2020 г. № 28);

- Методическими рекомендациями по проектированию дополнительных общеразвивающих программ (Письмо Минобрнауки РФ «О направлении информации» от 18 ноября 2015 г. N 09- 3242);

- Уставом муниципального бюджетного общеобразовательного учреждения «Остерская средняя школа».

**Направленность** - техническая.

**Актуальность программы** заключается во внедрении информационных технологий в разнообразные сферы деятельности, в том числе, как в учебную деятельность, так и физическое воспитание детей, на которых рассчитана данная программа.

**Педагогическая целесообразность программы** Данная дополнительная общеразвивающая программа направлена на развитие логического и пространственного мышления слушателя, способствует раскрытию творческого потенциала личности, формированию усидчивости и трудолюбия, приобретению практических умений и навыков в области компьютерных технологий, способствует интеллектуальному развитию ребенка и не требует специальных знаний, полученных до вступления в объединение.

**Учреждение (адрес):** муниципальное бюджетное общеобразовательное учреждение «Остерская средняя школа» (216537, Смоленская область, Рославльский район, село Остер, ул. Школьная, д.1

**Адресат программы:** Программа адресована детям от 11 до 15 лет всех категорий, в том числе детям с ОВЗ, инвалидам, детям, находящимся в трудной жизненной ситуации. Программа доступна для детей, проживающих в сельской местности, мотивированных детей.

**Сроки реализации программа** – одногодичная.

**Занятия проводятся** с группой 1 раз в неделю по 45 минут.

**Форма занятий:**

- индивидуальные;
- групповые;

- работа по командам;
- работа малыми группами.

**Цель программы:** ознакомление слушателя с объектно-ориентированным языком программирования Python, с возможностями, синтаксисом языка, технологией и методами программирования в среде Python, обучение практическим навыкам программирования на языке Python для решения типовых задач математики и информатики, а также при разработке простейших игр.

**Задачи программы:**

Личностные:

- создать условия для формирования навыков самоконтроля при работе за компьютером;
- создать условия для формирования навыков здорового образа жизни;
- создать условия для развития аккуратности, чувства вкуса, чувства гармонии, чувства цвета и композиции.

Метапредметные:

- развить мотивацию обучающихся к самостоятельному поиску информации;
- развить творческое воображение;
- развить умение применять полученные теоретические знания на практике.

Предметные:

- развитие интереса учащихся к изучению программирования;
- знакомство учащихся с основами программирования в среде Python;
- овладение базовыми понятиями теории алгоритмов при решении математических задач;
- формирование навыков работы в системе программирования Python;
- формирование самостоятельности и творческого подхода к решению задач с помощью средств современной вычислительной техники;
- формирование навыков алгоритмического и логического мышления;
- формирование навыков грамотной разработки программ;
- формирование практических навыков решения прикладных задач;
- формирование практических навыков разработки игр.

**Ожидаемые результаты освоения обучающимися программы кружка «Введение в язык программирования Python»**

В результате изучения дисциплины получают дальнейшее развитие личностные регулятивные, коммуникативные и познавательные универсальные учебные действия, учебная (общая и предметная) и общепользовательская ИКТ–компетентность обучающихся, составляющая психолого-педагогическую, инструментальную основу формирования способности и готовности к освоению систематических знаний, к их самостоятельному пополнению, переносу и интеграции, способности к

сотрудничеству и коммуникации, решению лично и социально значимых проблем и воплощению решений в практику, способности к самоорганизации, саморегуляции и рефлексии. В ходе изучения данного курса в основном формируются и получают развитие следующие метапредметные результаты:

- умение самостоятельно планировать пути достижения цели, в том числе
- альтернативные, осознанно выбирать наиболее эффективные способы решения задач;
- умение соотносить свои действия с планируемыми результатами, осуществлять
- контроль всей деятельности в процессе достижения результата, корректировать свои
- действия в соответствии с изменяющейся ситуацией;
- умение оценивать правильность выполнения учебной задачи, собственные
- возможности ее решения;
- владение основами самоконтроля, самооценки, принятия решений и осуществления осознанного выбора в учебной и познавательной деятельности;
- умение организовывать сотрудничество и совместную деятельность с учителем и сверстниками; работать индивидуально в группе: находить общее решение и разрешать конфликты на основе согласования позиций и учета интересов; формулировать, аргументировать и отстаивать свое мнение;
- формирование и развитие компетентности в области использования информационно коммуникационных технологий.

Вместе с тем, вносятся существенный вклад в развитие личностных результатов:

- формирование ответственного отношения к обучению, готовности и способности обучающихся к саморазвитию и самообразованию на основе мотивации к обучению и познанию;

#### Учебный план

№ п/п	Название раздела, темы	Количество часов			Формы контроля/ аттестации
		всего	теория	практика	
1.	История создания языка	1	1	0	Первичная диагностика.
2.	Установка Python	1	0	1	Тестирование
3.	Структура программы	1	0	1	Текущий контроль. Самостоятельная работа
4.	Линейные алгоритмы	3	1	2	Текущий контроль. Самостоятельная работа
5.	Алгоритмы с ветвлением	3	1	2	Текущий контроль. Самостоятельная

					работа
6.	Циклы	3	1	2	Текущий контроль. Самостоятельная работа
7.	Списки	3	1	2	Текущий контроль. Самостоятельная работа
8.	Функции	3	1	2	Текущий контроль. Самостоятельная работа
9.	Модули	3	1	2	Текущий контроль. Самостоятельная работа
10.	Работа с текстовыми файлами	5	2	3	Текущий контроль. Самостоятельная работа
11.	Графический модуль PyTurtle	5	2	3	Текущий контроль. Самостоятельная работа
12/	Создание приложения Painter	5	2	3	Защита проектов
	Итого:	36	13	23	

## Содержание программы

### Раздел 1. История создания языка

**Теория.** Введение: структура дисциплины. Основы техники безопасности и противопожарной безопасности, правила поведения в компьютерном кабинете. Профилактика нарушений здоровья при работе за компьютером. Права и обязанности членов объединения. Правила внутреннего распорядка. Организационные вопросы. Организация рабочего места. Стартовая диагностика обучающихся. Знакомство с языком программирования Python. История создания.

### Раздел 2. Установка Python

**Теория.** Python - это интерпретируемый язык программирования. Python - это полноценный язык программирования. Python – свободно распространяемый язык программирования.

**Практика.** Установка языка Python. Интерактивный режим. Создание первой программы: «Hello, world» текст.

### Раздел 3. Структура программы

**Теория.** Ввод и вывод данных. Операции. Создание и редактирование программ в среде IDLE. Типы данных: целые числа, числа с плавающей точкой, строки. Запуск программ.

**Практика.** Создать учебную задачу для соседа на ввод/вывод в Python или на последовательность выполнения арифметических операций.

### Раздел 4. Линейные алгоритмы.

**Теория.** Блок-схема линейного алгоритма. Ввод данных с клавиатуры. Правила записи арифметических выражений. Множественное присваивание. Случайные числа. Порядок выполнения операций. Вывод на экран.

**Практика.** Работа в парах: записать выражение по правилам языка Python и предложить напарнику вычислить его, не используя интерпретатора Python. Проверить себя в среде разработки.

### **Раздел 5. Алгоритмы с ветвлением.**

**Теория.** Инструкция ветвления if. Ветвление. Полная и сокращенная формы. Логический тип данных. Логическое высказывание. Операции сравнения. Множественное ветвление. Инструкция ветвления if ... else.

**Практика.** Программа «Открыто / закрыто».

### **Раздел 6. Циклы.**

**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.

**Практика.** Подсчет суммы цифр целого числа, решение задачи с циклом for, составление программ с циклом.

### **Раздел 7. Списки.**

**Теория.** Копирование списков. Линейный поиск. Сортировка списка.

**Практика.** Создать список учеников класса, отсортировать его и написать программу, которая спрашивает у пользователя число N и выводит фамилию и имя N-ого по алфавиту ученика, или N-ого с конца алфавита.

### **Раздел 8. Функции.**

**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.

**Практика.** Создание функций. Решение задач с использованием функций.

### **Раздел 9. Модули.**

**Теория.** Модули из стандартной библиотеки. Инструкция import. Инструкция from. Использование псевдонимов. Создание своего модуля.

**Практика.** Создание своего модуля.

### **Раздел 10. Работа с текстовыми файлами.**

**Теория.** Функция open и её аргументы. Чтение из файла. Запись в файл.

**Практика.** Создание игры «Викторина». Вопросы и варианты ответов с указанием правильного хранятся в текстовом файле. Вопросы задаются последовательно. Игроку предлагается выбрать правильный ответ из предложенных вариантов. В случае правильного ответа результат увеличивается на единицу, и игроку выводится сообщение: «верно». В случае неверного ответа выводится сообщение: «не верно». После ответа на

последний вопрос на экран выводится результат (количество верных ответов).

### **Раздел 11. Графический модуль PyTurtle**

**Теория.** Положение «черепашки». Координатная плоскость. Направление «черепашки». Команда поставить точку. Циклические конструкции в PyTurtle. Графический модуль PyTurtle. Создание геометрических фигур в PyTurtle. Рисование спиралей в PyTurtle. Конструкции ветвления алгоритмов в PyTurtle.

**Практика.** Рисуем снежинки. Конкурс на самую необычную снежинку.

### **Раздел 12. Создаем приложения Painter.**

**Теория.** Импорт модуля tkinter. Создание базового окна. Создание рамки. Создание метки. Создание кнопок. Менеджер размещения Grid. Текстовые поля и области. Применение флажков. Применение переключателей.

**Практика.** Создаем приложение «Painter», позволяющее рисовать мышкой на экране (с палитрой цветов и ластиком).

### **Календарный учебный график**

№ п/п	Месяц	Форма занятия	Количество часов	Тема занятия	Место проведения	Форма контроля
1.	сентябрь	лекция и инструктаж	1	Вводное занятие. Правила безопасного поведения в компьютерном классе	кабинет	опрос
2.	сентябрь	индивидуальная	1	Что такое Python. Установка Python.	кабинет	контрольное задание
3.	сентябрь	индивидуальная	1	Ввод и вывод данных. редактирование программ в среде IDLE. Запуск программ.	кабинет	контрольные вопросы
4.	сентябрь	индивидуальная	1	Блок-схема линейного алгоритма. Ввод данных с клавиатуры.	кабинет	контрольное задание
5.	октябрь	индивидуальная	1	Правила записи арифметических выражений. Множественное присваивание.	кабинет	опрос
6.	октябрь	индивидуальная	1	Случайные числа. Порядок выполнения операций.	кабинет	опрос
7.	октябрь	индивидуальная	1	Инструкция ветвления if. Полная и сокращенная формы.	кабинет	опрос
8.	октябрь	индивидуальная	1	Логический тип данных. Логическое высказывание. Операции сравнения.	кабинет	контрольное задание
9.	октябрь	индивидуальная	1	Множественное ветвление.	кабинет	контрольное

		ьная		Инструкция ветвления if ... else.		задание
10.	ноябрь	индивидуал ьная	1	Понятие цикла. Тело цикла. Условия выполнения тела цикла. Оператор цикла с условием.	кабинет	контрольное задание
11.	ноябрь	индивидуал ьная	1	Оператор цикла while. Бесконечные циклы. Альтернативная ветка цикла while.	кабинет	контрольное задание
12.	ноябрь	индивидуал ьная	1	Оператор цикла с параметром for. Примеры решения задач с циклом.	кабинет	контрольное задание
13.	ноябрь	индивидуал ьная	1	Копирование списков.	кабинет	контрольное задание
14.	декабрь	индивидуал ьная	1	Линейный поиск.	кабинет	контрольное задание
15.	декабрь	индивидуал ьная	1	Сортировка списка.	кабинет	контрольное задание
16.	декабрь	индивидуал ьная	1	Создание функций. Параметры и аргументы. Локальные и глобальные переменные.	кабинет	контрольное задание
17.	декабрь	индивидуал ьная	1	Поток выполнения. Функции, возвращающие результат.	кабинет	контрольное задание
18.	январь	индивидуал ьная	1	Примеры решения задач с использованием функций.	кабинет	контрольное задание
19.	январь	индивидуал ьная	1	Модули из стандартной библиотеки.	кабинет	контрольное задание
20.	январь	индивидуал ьная	1	Инструкция import. Инструкция from. Использование псевдонимов.	кабинет	контрольное задание
21.	январь	индивидуал ьная	1	Создание своего модуля.	кабинет	контрольное задание
22.	февраль	индивидуал ьная	1	Функция open и её аргументы.	кабинет	
23.	февраль	индивидуал ьная	1	Чтение из файла.	кабинет	опрос
24.	февраль	индивидуал ьная	1	Запись в файл.	кабинет	контрольное задание
25.	февраль	индивидуал ьная	1	Запись в файл.	кабинет	опрос
26.	март	индивидуал ьная	1	Запись в файл.	кабинет	тестирование
27.	март	индивидуал ьная	1	Положение «черепашки». Координатная плоскость.	кабинет	презентация своей работы
28.	март	индивидуал ьная	1	Циклические конструкции в PyTurtle.	кабинет	опрос
29.	март	индивидуал ьная	1	Графический модуль PyTurtle.	кабинет	презентация своей работы



30.	апрель	индивидуальная	1	Создание геометрических фигур в PyTurtle.	кабинет	опрос
31.	апрель	индивидуальная	1	Рисование спиралей в PyTurtle.	кабинет	тестирование
32.	апрель	индивидуальная	1	Импорт модуля tkinter. Создание базового окна, рамки, метки, кнопок.	кабинет	опрос
33.	май	индивидуальная	1	Менеджер размещения Grid. Текстовые поля и области.	кабинет	контрольное задание
34.	май	индивидуальная	1	Создаем приложение «Painter»	кабинет	контрольное задание
35.	май	индивидуальная	1	Создаем приложение «Painter»	кабинет	контрольное задание
36.	май	индивидуальная	1	Создаем приложение «Painter»	кабинет	контрольное задание

### Методическое обеспечение программы

Целостная включенность в образовательный процесс возможна только при эмоциональной реакции ученика. Постигание компьютера должно идти через постижение самого себя, собственной уникальности в частности, и уникальности человека, в общем.

Решение всех выше названных проблем возможно с развитием творческого потенциала личности ребенка. Творчество необходимо рассматривать как процесс, а не как продукт. Необходимо быть постоянно изменяющимся в изменяющемся мире.

Невозможно дать всю сумму знаний, который будет необходим в дальнейшей жизни. И становится актуальным развитие таких качеств личности как гибкость, адаптивность, восприимчивость мышления, идущего в разных направлениях, как в основной, учебной деятельности, так и в социальной сфере.

Воспитывающий эффект программы заключается в том, что каждый выбирает для себя посильную задачу, предполагая, что для решения простой задачи необходимо потратить меньшее количество усилий и поощрение будет соответствующим.

Каждый выбирает для себя вершину сам и это его собственная вершина, на которой нет конкурентов, не надо никого толкать, и никто тебя не толкает.

Программа реализуется в ознакомительном плане в силу её сложности и опирается на практическую работу учащихся.

#### Формы проведения занятий:

- индивидуальная работа;
- творческие задания;

- практическая работа;
- дискуссия;
- обучающие игры (имитации и образовательные игры);
- изучение и закрепление нового материала на интерактивной лекции (лекция-беседа, лекция-дискуссия, лекция с разбором конкретных ситуаций, лекция с заранее запланированными ошибками, лекция пресс-конференция, мини-лекция);
- эвристическая беседа;
- просмотр и обсуждение видеofilьмов.

### **Материально- техническое оснащение программ**

Продуктивность работы во многом зависит от качества материально-технического оснащения процесса, инфраструктуры организации и иных условий. При реализации Программы используются методические пособия, дидактические материалы, материалы на электронных носителях.

Для успешного проведения занятий и выполнения программы в полном объеме необходимы:

#### **инфраструктура организации:**

- учебный кабинет Центра образования цифрового и гуманитарного профилей «Точка роста»;

#### **технические средства обучения:**

- ✓ ноутбуки – 10 шт. (операционная система Windows10);
- ✓ среда разработки: Python 3.11.0 и выше;

#### **Педагогические технологии, используемые в обучении.**

1. Технология проблемного диалога. Учащимся не только сообщаются готовые знания, но и организуется такая их деятельность, в процессе которой они сами делают «открытия», узнают что-то новое и используют полученные знания и умения для решения жизненных задач.

2. Технология коллективного взаимообучения («организованный диалог», «сочетательный диалог», «коллективный способ обучения (КСО), «работа учащихся в парах сменного состава») позволяет плодотворно развивать у обучаемых самостоятельность и коммуникативные умения.

3. Игровая технология. Игровая форма в образовательном процессе создаётся при помощи игровых приёмов и ситуаций, выступающих как средство побуждения к деятельности. Способствует развитию творческих способностей, продуктивному сотрудничеству с другими учащимися. Приучает к коллективным действиям, принятию решений, учит руководить и подчиняться, стимулирует практические навыки, развивает воображение.

4. Элементы здоровьесберегающих технологий являются необходимым условием снижения утомляемости и перегрузки учащихся.

5. Проектная технология предлагает практические творческие задания, требующие от учащихся их применение для решения проблемных заданий, знания материала на данный исторический этап. Овладевая культурой проектирования, школьник приучается творчески мыслить, прогнозировать возможные варианты решения стоящих перед ним задач.

6. Информационно-коммуникационные технологии активизируют творческий потенциал учащихся; способствует развитию логики, внимания, речи, повышению качества знаний; формированию умения пользоваться информацией, выбирать из нее необходимое для принятия решения, работать со всеми видами информации, программным обеспечением, специальными программами и т.д.

### Формы аттестации (контроля)

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.

## Приложение 1

### Входящий контроль

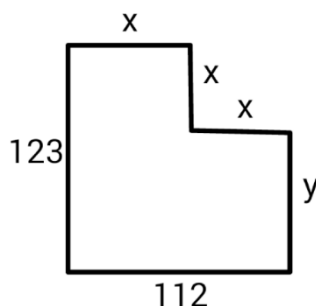
Низкий уровень: 0-70% выполненных заданий;

Средний уровень: 70-85% правильно выполненных заданий;

Высокий уровень: 85-100% правильно выполненных заданий.

<b>Фамилия, Имя</b>	
---------------------	--

1. Из квадрат со стороной  $x$ , рисунке. Чему равна



прямоугольника вырезали получилась фигура как на 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.

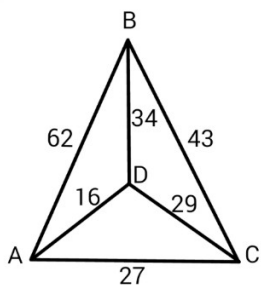
Сколько кусков рулета получится, если разрезать по кольцам всех трёх цветов?

Примечания

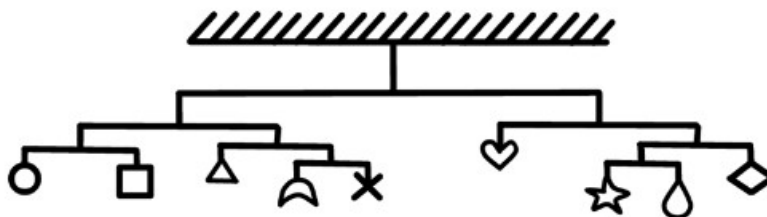
Учтите, что наклеить кольцо одного цвета на кольцо другого нельзя.

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?

## Приложение 2

Итоговый контроль

Диагностика

Промежуточ-ная диагностика

Итоговая

диагностика

№/ 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)

<p>Составьте выражение для вычисления в интерпретаторе Python 3 и вставьте в поле ответа результат вычисления:</p> <p style="text-align: center;">11111·1111111</p> <p>— <b>произведение чисел 11111 (5 единиц) и 1111111 (7 единиц)</b></p>
<p>Запишите число <b>1.2345e3</b> в виде <b>десятичной</b> дроби.</p>
<p>Составьте и запишите выражение для вычисления:</p> <p style="text-align: center;"><b>2014.0^14(Возвестив 14 степень)</b></p> <p>Обратите внимание на запись числа: это вещественное число.</p>
<p><b>Приведите к целому типу число 2.99</b></p>
<p>Расставьте скобки в выражении</p> <p style="text-align: center;"><b>a and b or not a and not b</b></p> <p>в соответствии с порядком вычисления выражения (приоритетом операций). Всего потребуется 5 пар скобок (внешние скобки входят в их число).</p>
<p>Найдите результат выражения для заданных значений <i>a</i> и <i>b</i> Учитывайте регистр символов при ответе.</p> <p><b>a = True</b> <b>b = False</b> <b>a and b or not a and not b</b></p>
<p>Отметьте выражения, значения которых равны True:</p> <ul style="list-style-type: none"><li>• "239" &lt; "30" and 239 &lt; 30</li><li>• "239" &lt; "30" and 239 &gt; 30</li><li>• "239" &gt; "30" and 239 &lt; 30</li><li>• "239" &gt; "30" and 239 &gt; 30</li></ul>
<p>Укажите результат выражения: "123" + "42"</p>
<p>Какое значение будет у переменной <i>i</i> после выполнения фрагмента программы?</p> <pre><b>i = 0</b> <b>while i &lt;= 10:</b> <b>    i = i + 1</b> <b>    if i &gt; 7:</b> <b>        i = i + 2</b></pre>
<p>Сколько итераций цикла будет выполнено в этом фрагменте программы?</p> <pre><b>i = 0</b> <b>while i &lt;= 10:</b> <b>    i = i + 1</b> <b>    if i &gt; 7:</b> <b>        i = i + 2</b></pre>
<p>Сколько всего знаков * будет выведено после исполнения фрагмента</p>

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.**

1. 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.

2. 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  $\pi$  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

3. 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.

4. 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.

5. 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.

6. Напишите программу, которая выводит часть последовательности 1 2 2 3 3 3 4 4 4 4 5 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  $n=5$ )

### **Список литературы.**

1. Python. Подробный справочник Дэвида М. Бизли — книга со справочной информацией о языке Python и модулях стандартной библиотеки.
2. Python. Справочник Марка Лутца. Справочник по наиболее часто используемым функциям и модулям.

### **Интернет-ресурсы**

1. Домашняя страница Python [www.python.org](http://www.python.org) . Справочные материалы, официальная документация.
2. Сайт проекта Интуит: Национальный открытый университет, курс «Введение в программирование на Python», <http://www.intuit.ru/studies/courses/12179/1172/info>
3. Сайт проекта Интуит: Национальный открытый университет. Курс «Язык программирования Python» <http://www.intuit.ru/studies/courses/49/49/info>
4. Сайт проекта Open Book Project [openbookproject.net](http://openbookproject.net) содержит серию практических примеров на Python Криса Мейерса.