**МИНИCTEPCTBO НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное автономное**

**образовательное учреждение высшего образования**

**«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»**

**Кафедра инфокоммуникаций**

**Отчет по лабораторной работе №4**

**Условные операторы и циклы в языке Python 3.**

**по дисциплине «Основы кроссплатформенного программирования»**

|  |
| --- |
| Выполнил студент группы ИВТ-б-о-20-1 |
| Токарев В. А. « »\_\_\_\_\_\_\_\_\_20\_\_г. |
| Подпись студента\_\_\_\_\_\_\_\_\_\_\_ |
| Работа защищена « »\_\_\_\_\_\_\_\_\_\_\_\_20\_\_г. |
| Проверил Воронкин Р. А. \_\_\_\_\_\_\_\_  (подпись) |

Ставрополь 2021

**Цель работы:** приобретение навыков программирования

разветвляющихся алгоритмов и циклических структур. Освоить операторы if,

while, for, continue.

**Ход работы:**

Ссылка на репозиторий: <https://github.com/UnDeR-The-mAsK/lab4.git>

1. Проработал примеры из методички.

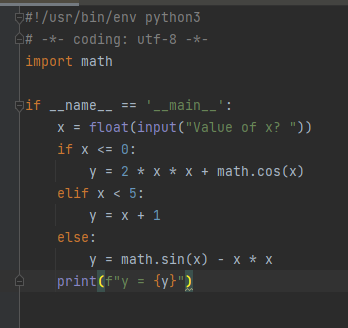


Рисунок 1. Код первого примера

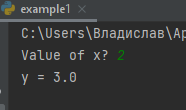


Рисунок 2. Результат выполнения программы

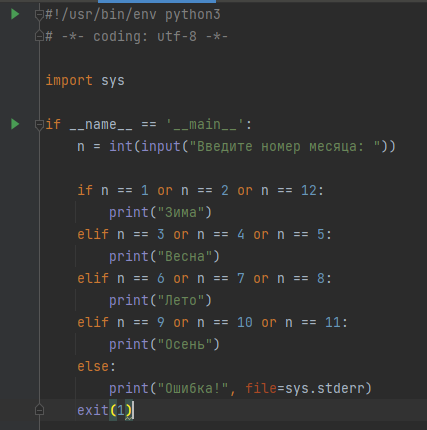


Рисунок 3. Код второго примера

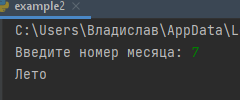


Рисунок 4. Результат выполнения программы

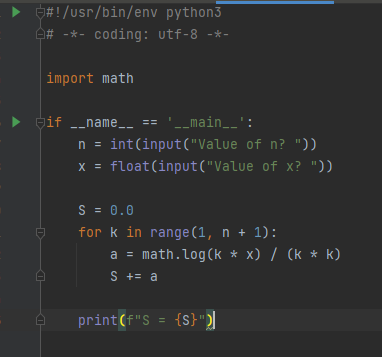


Рисунок 5. Код третьего примера

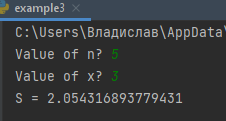


Рисунок 6. Работа программы третьего примера



Рисунок 7. Код четвертого примера

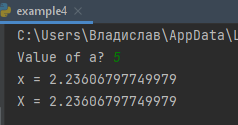


Рисунок 8. Работа кода четвертого примера

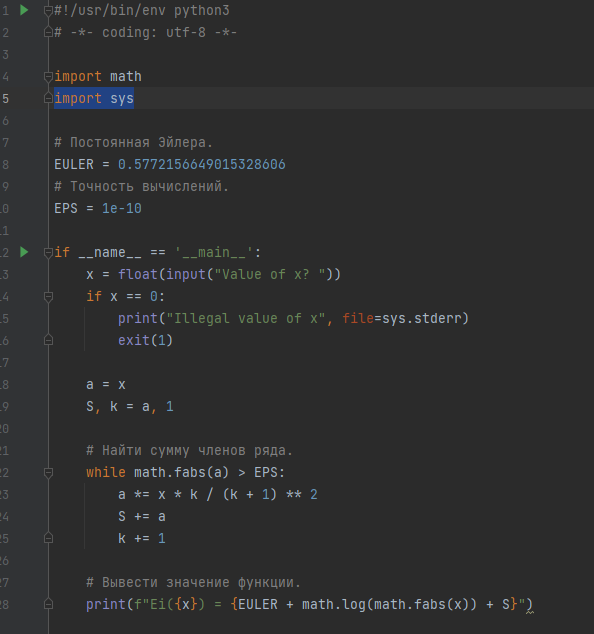


Рисунок 9. Код пятого примера

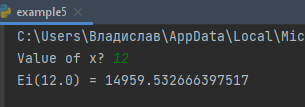


Рисунок 10. Работоспособность кода пятого примера

**Индивидуальное задание № 1.**

**Вариатн 5**

**Условие:** Организуйте свой репозиторий в соответствие с моделью ветвления git-flow.

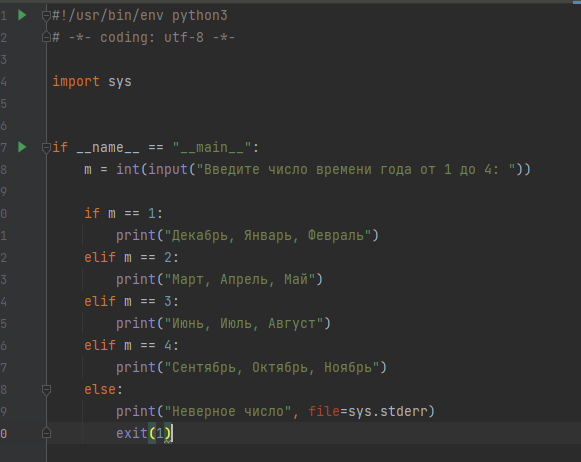


Рис. Индивидуальное задание 1

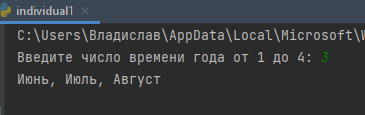


Рис. Результат инд. задания 1

**Индивидуальное задание № 2.**

**Вариатн 17**

**Условие:** Определить, есть ли среди трёх заданных чисел нечётные.

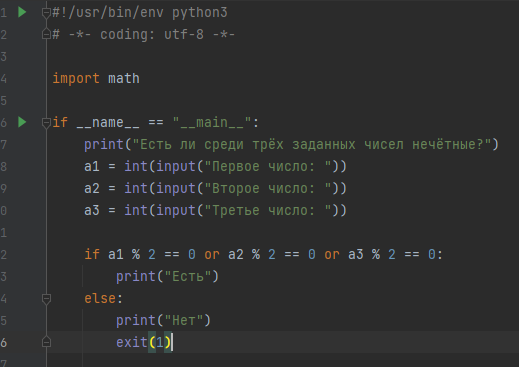


Рис. Индивидуальное задание 2

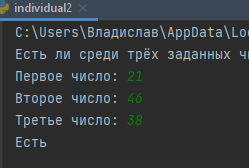


Рис. Результат инд. задания 2

**Индивидуальное задание № 3.**

**Вариатн 16**

**Условие:** Ученик выучил в первый день 5 английских слов. В каждый следующий день он выучивал на 2 слова больше, чем в предыдущий. Сколько английских слов выучит ученик в 10-ый день занятий

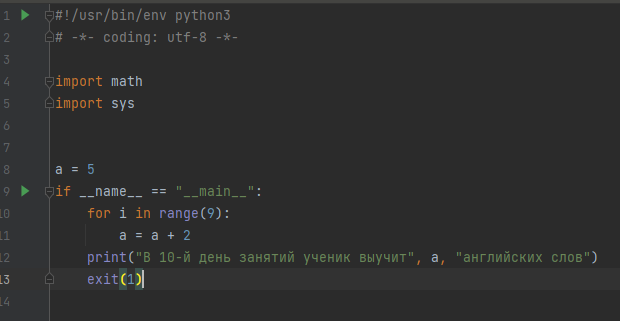


Рис. Индивидуальное задание 3

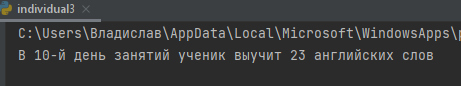


Рис. Результат инд. задания 3

**Контрольные вопросы:**

1. Для чего нужны диаграммы деятельности UML?

Диаграмма UML — наглядное представление совокупности элементов

модели в виде графа. Диаграммы UML применяются для визуализации разных

аспектов устройства или поведения моделируемой системы. UML пригоден

для моделирования любых систем: от информационных систем масштаба

предприятия до распределенных Web-приложений.

2. Что такое состояние действия и состояние деятельности?

Состояние действия - операции, проводимые над объектами или

различные вычисления.

В противоположность этому состояния деятельности могут быть

подвергнуты дальнейшей декомпозиции, вследствие чего выполняемую

деятельность можно представить с помощью других диаграмм деятельности.

Состояния деятельности не являются атомарными, то есть могут быть

прерваны. Предполагается, что для их завершения требуется заметное время.

Можно считать, что состояние действия - это частный вид состояния

деятельности, а конкретнее – такое состояние, которое не может быть

подвергнуто дальнейшей декомпозиции. А состояние деятельности можно

представлять себе, как составное состояние, поток управления которого

включает только другие состояния деятельности и действий.

3. Какие нотации существуют для обозначения переходов и ветвлений

в диаграммах деятельности?

Когда действие или деятельность в некотором состоянии завершается,

поток управления сразу переходит в следующее состояние действия или

деятельности. Для описания этого потока используются переходы,

показывающие путь из одного состояния действия или деятельности в другое.

В UML переход представляется простой линией со стрелкой

4. Какой алгоритм является алгоритмом разветвляющейся структуры?

Алгоритм разветвляющейся структуры - это алгоритм, в котором

вычислительный процесс осуществляется либо по одной, либо по другой

ветви, в зависимости от выполнения некоторого условия.

5. Чем отличается разветвляющийся алгоритм от линейного?

Линейный идет лишь в одном направлении, а в разветвляющемся

возможны разные исходы.

6. Что такое условный оператор? Какие существуют его формы?

Условные операторы: if и while. Условный оператор позволяет

выполнять действия в зависимости от логического значения условия.

7. Какие операторы сравнения используются в Python?

Операторы сравнения: > (больше), < (меньше), >= (больше или равно),

<= (меньше или равно), = = (равно), != (не равно).

8. Что называется простым условием? Приведите примеры.

Простое условие - два выражения,связанные одним из знаков

отношений. If a>4: print(‘Hello’)

9. Что такое составное условие? Приведите примеры.

Составные условия - условия, состоящие из двух или более простых

условий, соединенных с помощью логических операций and (и), or (или), not

(не).

10. Какие логические операторы допускаются при составлении сложных

условий?

В сложных условиях можно использовать операторы and (и), or (или),

not (не).

11. Может ли оператор ветвления содержать внутри себя другие

ветвления?

Да операторы ветвления могу иметь внутри другие ветвления в виде

множественного ветвления

12. Какой алгоритм является алгоритмом циклической структуры?

Оператор for выполняет указанный набор инструкций заданное

количество раз, которое определяется количеством элементов в наборе.

13. Типы циклов в языке Python.

Существует 2 вида циклов: while- Оператор цикла while выполняет

указанный набор инструкций до тех пор, пока условие цикла истинно. For-

Оператор for выполняет указанный набор инструкций заданное количество

раз, которое определяется количеством элементов в наборе.

14. Назовите назначение и способы применения функции range.

Функция range возвращает неизменяемую последовательность чисел в

виде объекта range. Функцию range можно использовать для генерации

последовательности чисел.

15. Как с помощью функции range организовать перебор значений от 15

до 0 с шагом 2?

list(range(15, 0, 2))

16. Могут ли быть циклы вложенными?

Python позволяет вкладывать циклы друг друга. Вложенный цикл – это

цикл, который встречается внутри другого цикла.

17. Как образуется бесконечный цикл и как выйти из него?

Бесконечный цикл while — это цикл, в котором условие никогда не

становится ложным. Это значит, что тело исполняется снова и снова, а цикл

никогда не заканчивается. Прервать его можно с помощью break и continue

18. Для чего нужен оператор break.

Оператор break предназначен для досрочного прерывания работы цикла

while.

19. Для чего нужен оператор continue?

Оператор continue запускает цикл заново, при этом код, расположенный

после данного оператора, не выполняется.

20. Для чего нужны стандартные потоки stdout и stderr?

В операционной системе по умолчанию присутствуют стандартных

потока вывода на консоль: буферизованный поток stdout для вывода данных и информационных сообщений, а также небуферизованный поток stderr для

вывода сообщений об ошибках. По умолчанию функция print использует

поток stdout. Для того, чтобы использовать поток stderr необходимо передать

его в параметре file функции print. Само же определение потоков stdout и stderr находится в стандартном пакете Python sys. Хорошим стилем

программирования является наличие вывода ошибок в стандартный поток

stderr поскольку вывод в потоки stdout и stderr может обрабатываться как

операционной системой, так и сценариями пользователя по-разному.

21. Как в Python организовать вывод в стандартный поток stderr?

Для того, чтобы использовать поток stderr необходимо передать его в

параметре file функции print.

22. Каково назначение функции exit ?

В Python завершить программу и передать операционной системе

заданный код возврата можно посредством функции exit.

**Вывод:** в процессе выполнения лабораторной работы приобрел навыки

программирования разветвляющихся алгоритмов и циклических структур.

Освоил операторы if, while, for, continue.