

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО МГТУ «СТАНКИН»)**

|  |  |
| --- | --- |
| **Институт**  информационных технологий | **Кафедра**  информационных систем |

**Отчет по лабораторной работе №2**

по дисциплине «**Веб-программирование**»

на тему: «**Основы языка python**»

|  |  |  |
| --- | --- | --- |
| **Студент** группа ИДБ–19–05 |  | **Никитина А.С.** |
|  | подпись |  |
| Руководитель |  | **Адамова Ю.С.** |
|  | подпись |  |

**ВЫПОЛНЕНИЕ ЗАДАНИЙ**

Ссылка на github: <https://github.com/LinaWandel/Laboratornaya-2>

1. **ЗАДАНИЕ 1**

Написать функцию, которая на вход принимает int и возвращает true или false в зависимости является ли это число палиндром. Число является палиндромом, если оно читается справа налево и слева направо одинаково.

На рисунке 1.1 показана работа программы 1.

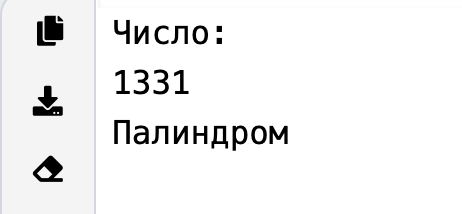


Рис.1.1. Работа программы 1

1. **ЗАДАНИЕ 2**

Написать функцию, которая принимает на вход список из положительных целочисленных элементов и возвращает три списка:

a. в первом - числа, которые делятся на 2

b. во втором - числа, которые делятся на 3

c. в третьем - числа, которые делятся на 5

На рисунке 2.1 показана работа программы 2.

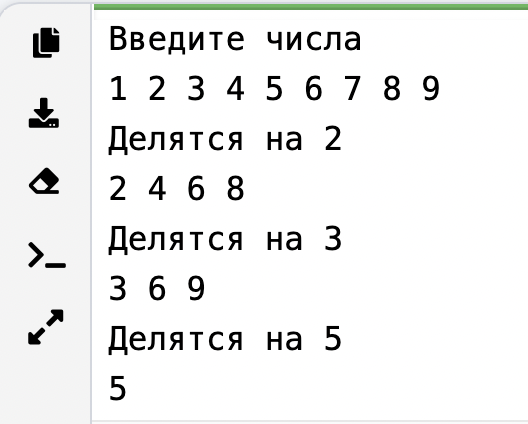


Рис.2.1. Работа программы 2

1. **ЗАДАНИЕ 3**

Написать функцию, принимающую на вход int, и число, обратное этому int.

На рисунке 3.1 показана работа программы 3.

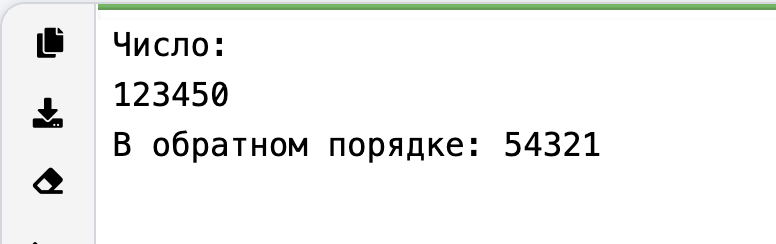


Рис.3.1. Работа программы 3

1. **ЗАДАНИЕ 4**

Написать функцию, которая будет расчитывать квадратный корень n-ой степени методом Ньютона.

На рисунке 4.1 показана работа программы 4.

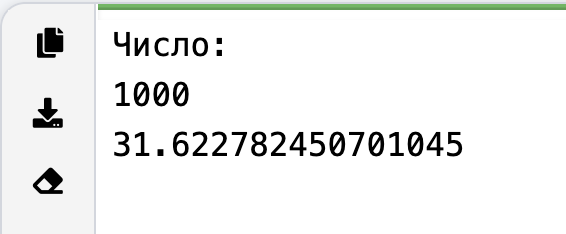


Рис.4.1. Работа программы 4

1. **ЗАДАНИЕ 5**

Написать функцию, принимающую 1 аргумент — число от 0 до 100000, и возвращающую true, если оно простое, false если нет.

На рисунке 5.1 показана работа программы 5.

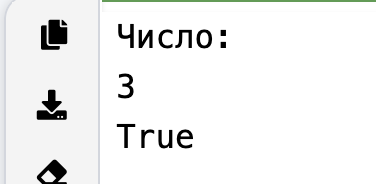


Рис.5.1. Работа программы 5

1. **ЗАДАНИЕ 7**

Написать декоратор, который будет кэшировать результат вызова функции и отдавать его при последующих вызовах данной функции (для тех, кто был на семинаре, но не обязательно - можете посмотреть как работают декораторы.

Усложненный вариант - написать тот же самый декоратор, но с параметром, который будет показывать сколько раз отдавать кешируемый результат. Если данный счетчик обнуляется, то выполняем функцию и вновь кешируем ее результат. На рисунке 6.1 показана работа программы 6.

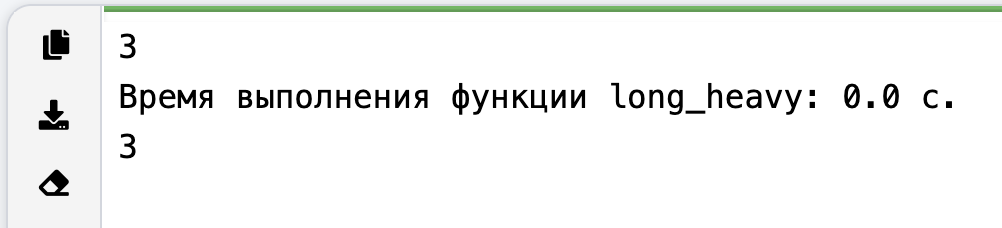


Рис.6.1. Работа программы 6

**ВЫВОД**

В ходе лабораторной работы были изучены основы языка Python.