

## Практическая работа №5

(Парфенов А.К. У-213)

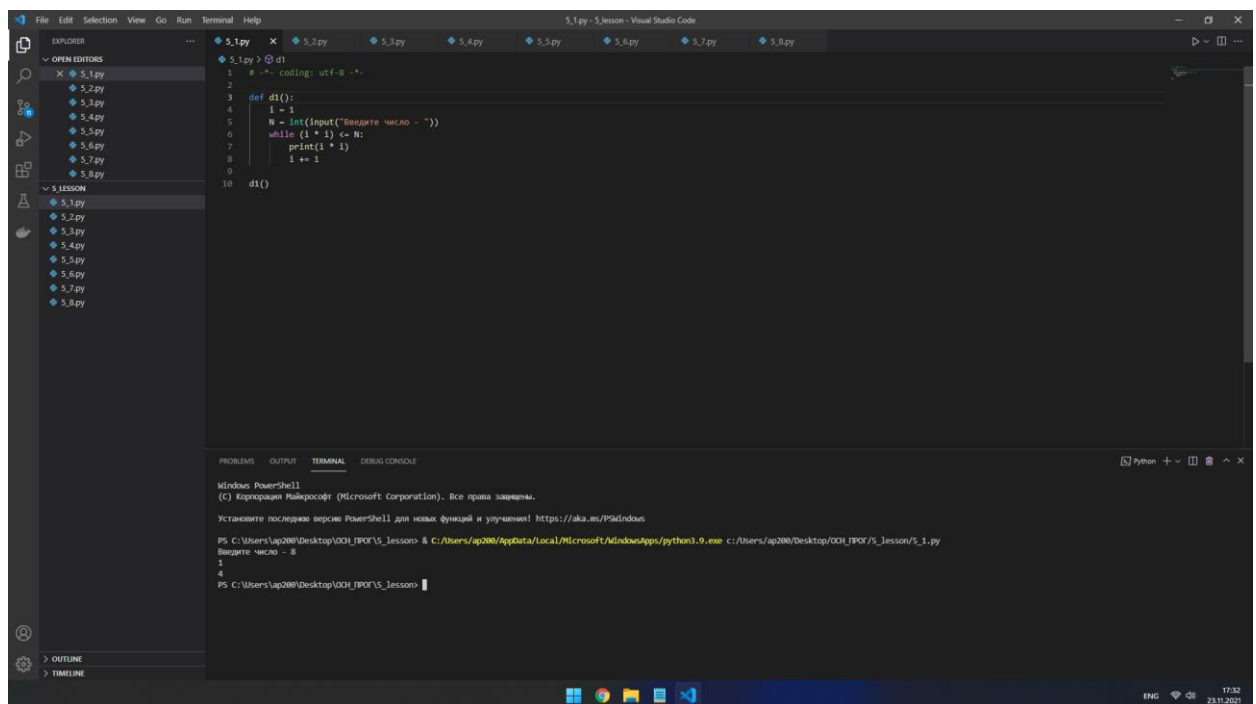
№1

1. По данному целому числу N распечатайте все квадраты натуральных чисел, не превосходящие N, в порядке возрастания.

```
# -*- coding: utf-8 -*-

def d1():
    i = 1
    N = int(input("Введите число - "))
    while (i * i) <= N:
        print(i * i)
        i += 1

d1()
```



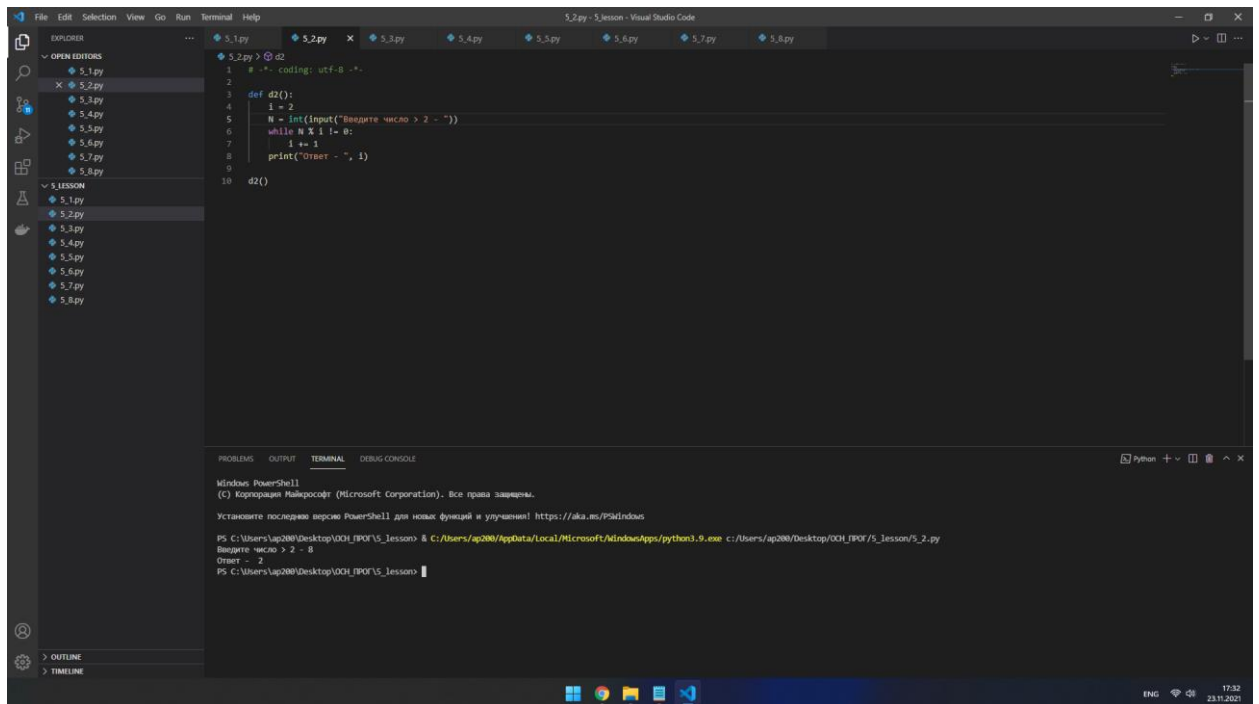
№2

1. Дано целое число, не меньшее 2. Выведите его наименьший натуральный делитель, отличный от 1.

```

2. # -*- coding: utf-8 -*-
3.
4. def d2():
5.     i = 2
6.     N = int(input("Введите число > 2 - "))
7.     while N % i != 0:
8.         i += 1
9.     print("Ответ - ", i)
10.
11. d2()

```



№3

1. По данному натуральному числу N найдите наибольшую целую степень двойки, не превосходящую N. Выведите показатель степени и саму степень. Операцией возведения в степень пользоваться нельзя!

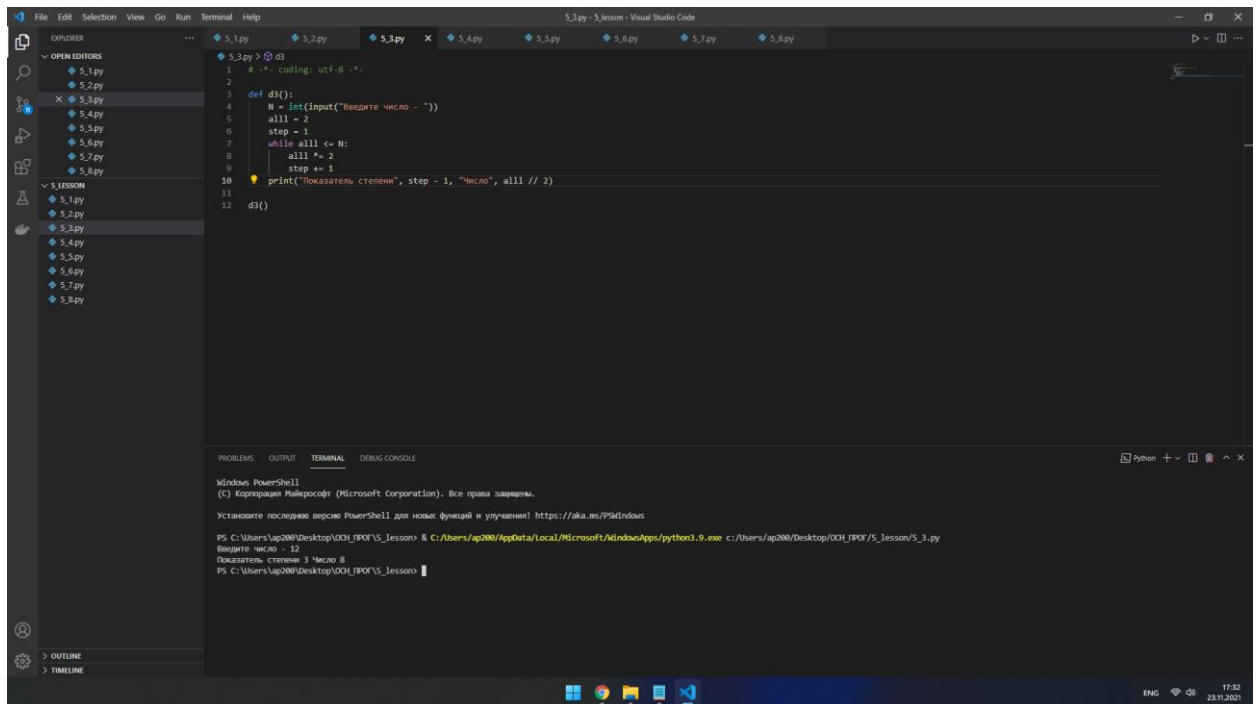
```

# -*- coding: utf-8 -*-

def d3():
    N = int(input("Введите число - "))
    all1 = 2
    step = 1
    while all1 <= N:
        all1 *= 2
        step += 1
    print("Показатель степени", step - 1, "Число", all1 // 2)

d3()

```



№4

1. В первый день спортсмен пробежал  $x$  километров, а затем он каждый день увеличивал пробег на 10% от предыдущего значения. По данному числу  $u$  определите номер дня, на который пробег спортсмена составит не менее  $u$  километров. Программа получает на вход действительные числа  $x$  и  $u$  и должна вывести одно натуральное число.

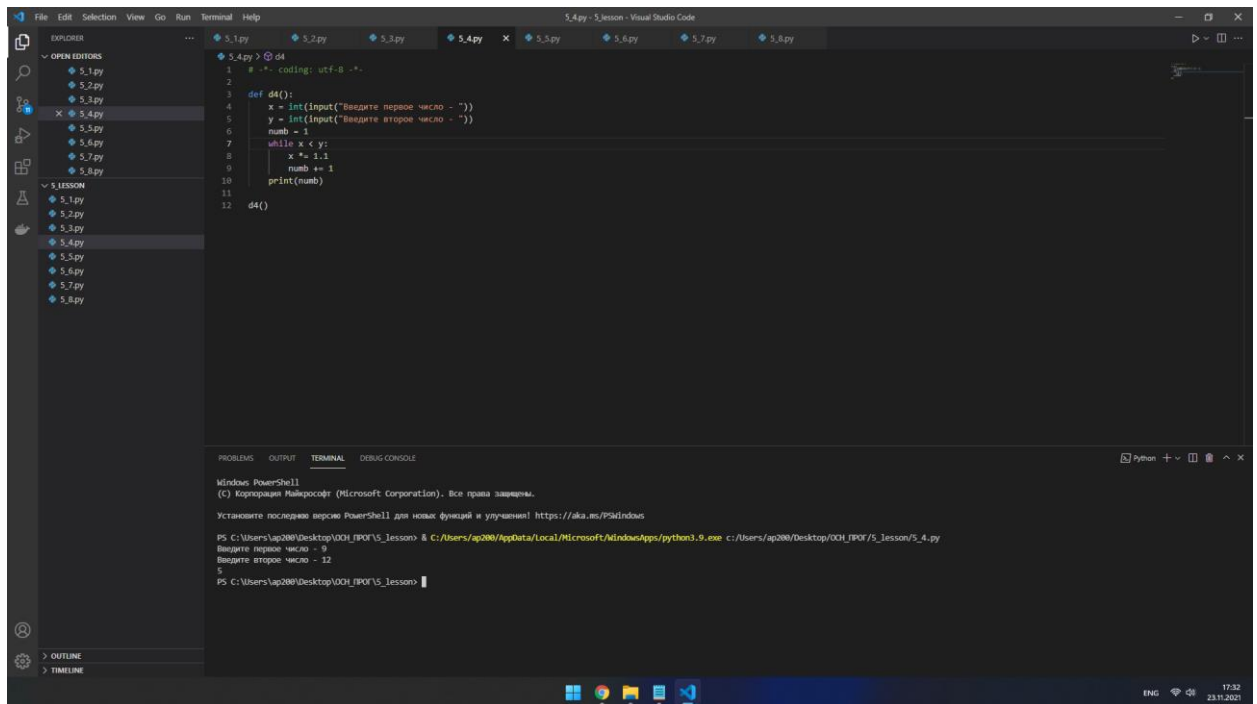
```

# -*- coding: utf-8 -*-

def d4():
    x = int(input("Введите первое число - "))
    y = int(input("Введите второе число - "))
    numb = 1
    while x < y:
        x *= 1.1
        numb += 1
    print(numb)

d4()

```



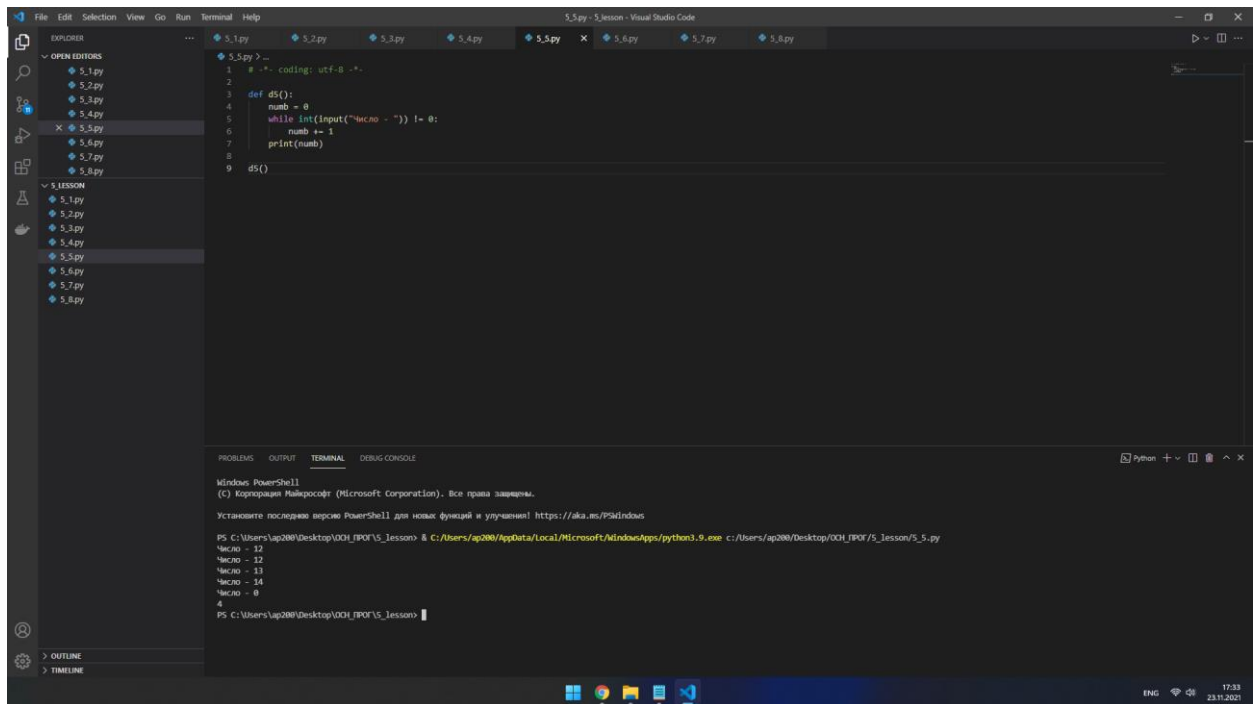
№5

1. Программа получает на вход последовательность целых неотрицательных чисел, каждое число записано в отдельной строке. Последовательность завершается числом 0, при считывании которого программа должна закончить свою работу и вывести количество членов последовательности (не считая завершающего числа 0). Числа, следующие за числом 0, считывать не нужно.

```
# -*- coding: utf-8 -*-

def d5():
    numb = 0
    while int(input("Число - ")) != 0:
        numb += 1
    print(numb)

d5()
```



№6

1. Определите среднее значение всех элементов последовательности, завершающейся числом 0.

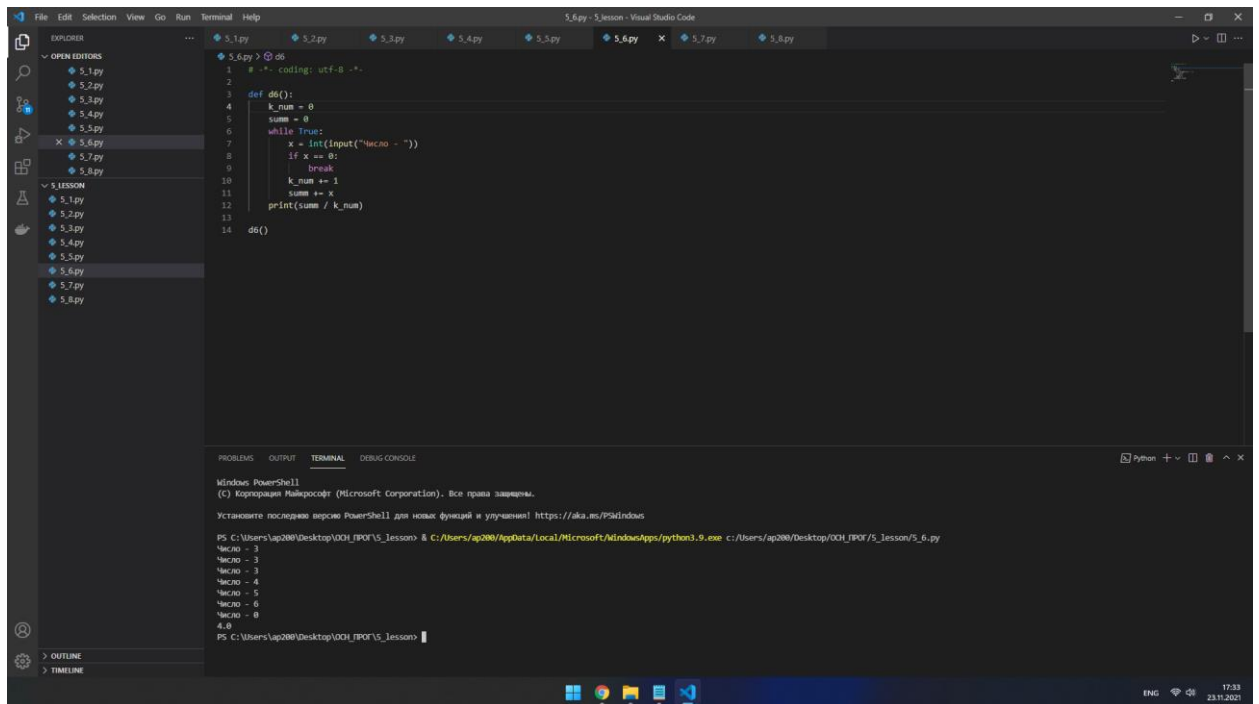
```

# -*- coding: utf-8 -*-

def d6():
    k_num = 0
    summ = 0
    while True:
        x = int(input("Число - "))
        if x == 0:
            break
        k_num += 1
        summ += x
    print(summ / k_num)

d6()

```



№7

1. Последовательность состоит из натуральных чисел и завершается числом 0. Определите, сколько элементов этой последовательности больше предыдущего элемента.

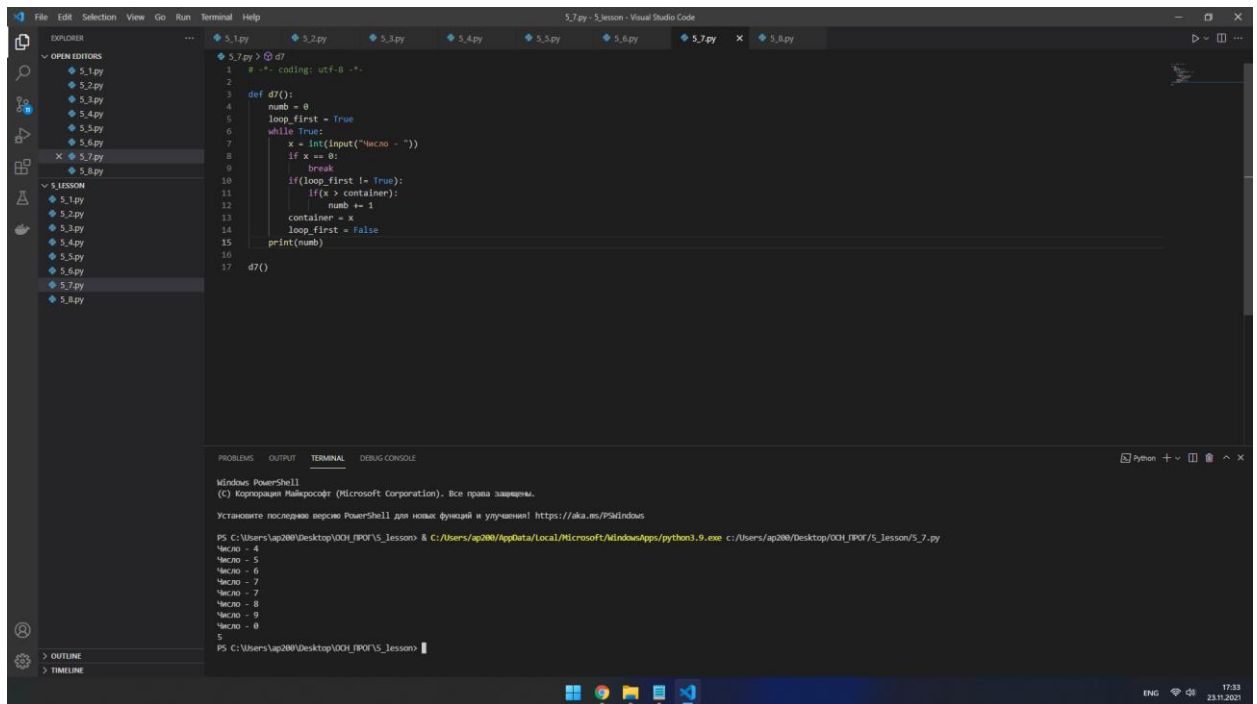
```

# -*- coding: utf-8 -*-

def d7():
    numb = 0
    loop_first = True
    while True:
        x = int(input("Число - "))
        if x == 0:
            break
        if(loop_first != True):
            if(x > container):
                numb += 1
        container = x
        loop_first = False
    print(numb)

d7()

```



№8

1. Дана последовательность натуральных чисел, завершающаяся числом 0. Определите, какое наибольшее число подряд идущих элементов этой последовательности равны друг другу.

```

2. # -*- coding: utf-8 -*-
3.
4. def d8():
5.     numb = 1
6.     maxStreak = 0
7.     loop_first = True
8.     while True:
9.         x = int(input("Введите число - "))
10.        if x == 0:
11.            break
12.        if(loop_first != True):
13.            if(x == container):
14.                numb += 1
15.            else:
16.                if(maxStreak < numb):
17.                    maxStreak = numb
18.                numb = 1
19.            container = x
20.        loop_first = False
21.        print(max(maxStreak, numb))
22.
23. d8()

```

