

## **Практическое занятие № 4**

**Тема:** составление программ циклической структуры в IDE PyCharm Community.  
Размещение проекта на GitHub.

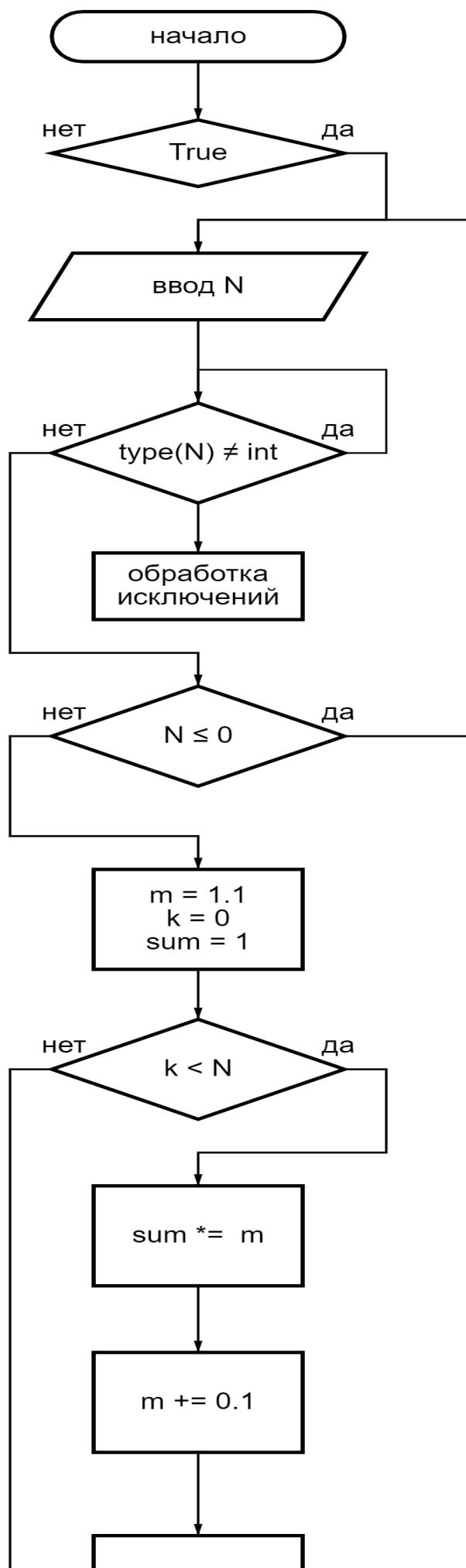
**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы  
составления программ, приобрести навыки составление программ циклической  
структурь в IDE PyCharm Community, первичные навыки работы с сервисом GitHub.

**Постановка задачи.**

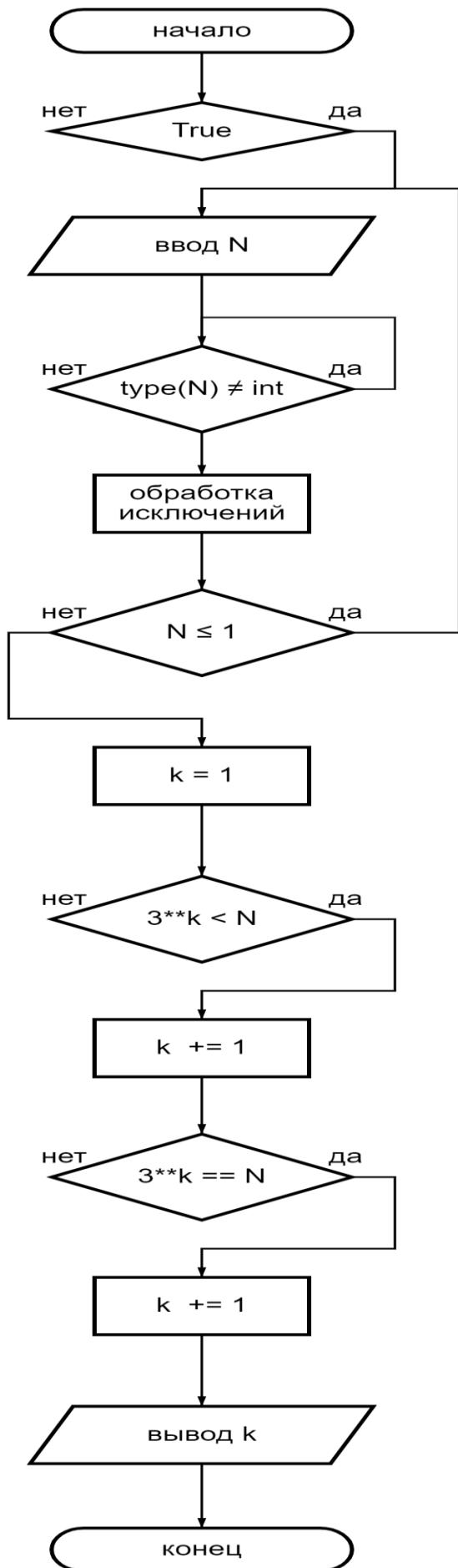
1. Разработать программу для нахождения произведения сомножителей выражения  
 $1.1 * 1.2 * 1.3 \dots$ , при этом пользователь должен вводить количество множителей.
2. Алгоритм нахождения наименьшего целого числа K, при котором  
выполняется неравенство  $3^{**}K > N$ , при этом  $N > 1$ .

**Тип алгоритма:** циклический.

**Блок-схема алгоритма:**



Блок-схема 2-ой программы:



### **Текст программы:**

#### **1. Текст 1-ой программы.**

```
# Алгоритм нахождения произведения сомножителей выражения 1.1 * 1.2
* 1.3...
while True:
    N = input("Введите кол-во множителей для выражения 1.1 * 1.2 *
1.3... в виде числа, которое больше 0: ")

    while type(N) != int: # Обработка исключений
        try:
            N = int(N)
        except ValueError:
            print("Вы не ввели число!")
            N = input("Введите кол-во множителей для выражения 1.1 *
1.2 * 1.3... в виде числа, которое больше 0: ")

    if N <= 0:
        print("Введите число, которое больше 0!")
        continue

    m = 1.1
    k = 0
    sum = 1
    while k < N:
        sum *= m
        m += 0.1
    k += 1
    break
print(sum)
```

#### **2. Текст 2-ой программы.**

```
# Алгоритм нахождения наименьшего целого числа K, при котором
# выполняется неравенство 3**K > N, при этом N > 1.

while True:
    N = input("Введите целое число, которое больше 1: ")

    while type(N) != int: # Обработка исключений
        try:
            N = int(N)
        except ValueError:
            print("Вы не ввели число!")
            N = input("Введите целое число, которое больше 1: ")

    if N <= 1:
        print("Введите число, которое больше 1!")
        continue
```

```
k = 1
while 3**k < N:
    k += 1
if 3**k == N:
    k += 1
print("При неравенстве 3**k > N, k = ", k)
break
```

### Протокол работы программы:

1. Протокол работы 1-ой программы:

Введите кол-во множителей для выражения 1.1 \* 1.2 \* 1.3... в виде числа, которое больше 0: 4

2.4024000000000014

Process finished with exit code 0

2. Протокол работы 2-ой программы:

Введите целое число, которое больше 1: 26

При неравенстве  $3^{**}k > N$ ,  $k = 3$

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал(а) навыки составления программ циклической структуры в IDE PyCharm Community. Были использованы языковые конструкции `while`, `if`.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода.

Готовые программные коды выложены на GitHub.