

Практическое задание № 4

**Тема:** составление программ циклической структуры в IDE PyCharm Community

**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ циклической структуры в IDE PyCharm Community.

**Постановка задачи №1.**

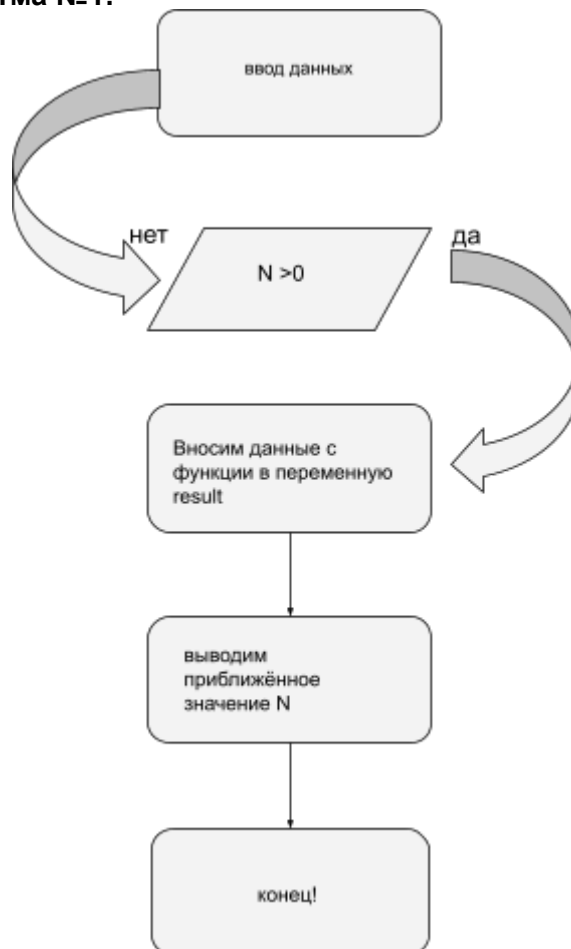
Разработать программу, которая получает число  $N$  ( $>0$ ). Используя один цикл, найти сумму  $1 + 1/(1!) + 1/(2!) + 1/(3!) + \dots + 1/(N!)$  (выражение  $N!$  —  $N$ -факториал — обозначает произведение всех целых чисел от 1 до  $N$ :  $N! = 1 \cdot 2 \cdot \dots \cdot N$ ). Полученное число является приближенным значением константы  $e = \exp(1)$ .

**Постановка задачи №2.**

Даны положительные числа  $A$  и  $B$  ( $A > B$ ). На отрезке длины  $A$  размещено максимально возможное количество отрезков длины  $B$  (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка  $A$ .

**Тип алгоритма:** циклический.

**Блок-схема алгоритма №1:**



**Блок-схема алгоритма №2:**



Студент группы ИС-27 Орешников И.Д.

### Текст программы №1:

```

#Дано целое число N (>0) .
# Используя один цикл, найти сумму 1 + 1/(1!) + 1/(2!) + 1/(3!) + ... + 1/(N!)
# (выражение N! – N-факториал – обозначает произведение всех целых чисел от 1 до N:
N! = 1·2·... · N).
# Полученное число является приближенным значением константы e = exp(1) .
try:
    def calculate_exp(N):
        factorial = 1
        exp = 1.0
        for i in range(1, N + 1):
            factorial *= i
            exp += 1.0 / factorial
        return exp
    # Введите значение N
    N = int(input("Введите целое число N (>0): "))
    if N > 0:
        result = calculate_exp(N)
        print(f"Приближенное значение e при N = {N} равно {result}")
except ValueError:
    print("Введите число больше нуля!!")
  
```

### Протокол работы №1:

Введите целое число N (>0): 4

Приближенное значение e при N = 4 равно 2.7083333333333333

Process finished with exit code 0

### Текст программы №2:

```
#Даны положительные числа A и B (A > B).  
#Отрезок длины A размещен на отрезке длины B. Определите максимально возможное количество  
# Не используя операции умножения и деления, найти длину незанятой  
# части отрезка A.  
def length(A, B):  
    if A <= B:  
        return 0 # Если A не больше B, то незанятой части нет  
  
    remaining_length = A  
    if remaining_length >= B:  
        remaining_length -= B  
  
    return remaining_length  
  
# Введите значения A и B  
try:  
    A = int(input("Введите положительное число A: "))  
    B = int(input("Введите положительное число B (меньше A): "))  
except ValueError:  
    print("Введите положительное число!")  
  
if A > 0 and B > 0 and A > B:  
    result = length(A, B)  
    print(f"Длина незанятой части отрезка A равна {result}")  
else:  
    print("Пожалуйста, введите положительные числа, где A > B.")
```

### Протокол работы №2:

Введите положительное число A: 100

Введите положительное число B (меньше A): 10

Длина незанятой части отрезка A равна 90

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ цикличной структуры в IDE PyCharm Community. Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.