

Практические занятия №4

Тема: составление программ циклической структуры в IDE PyCharm Community.

Размещение проекта на GitHub.

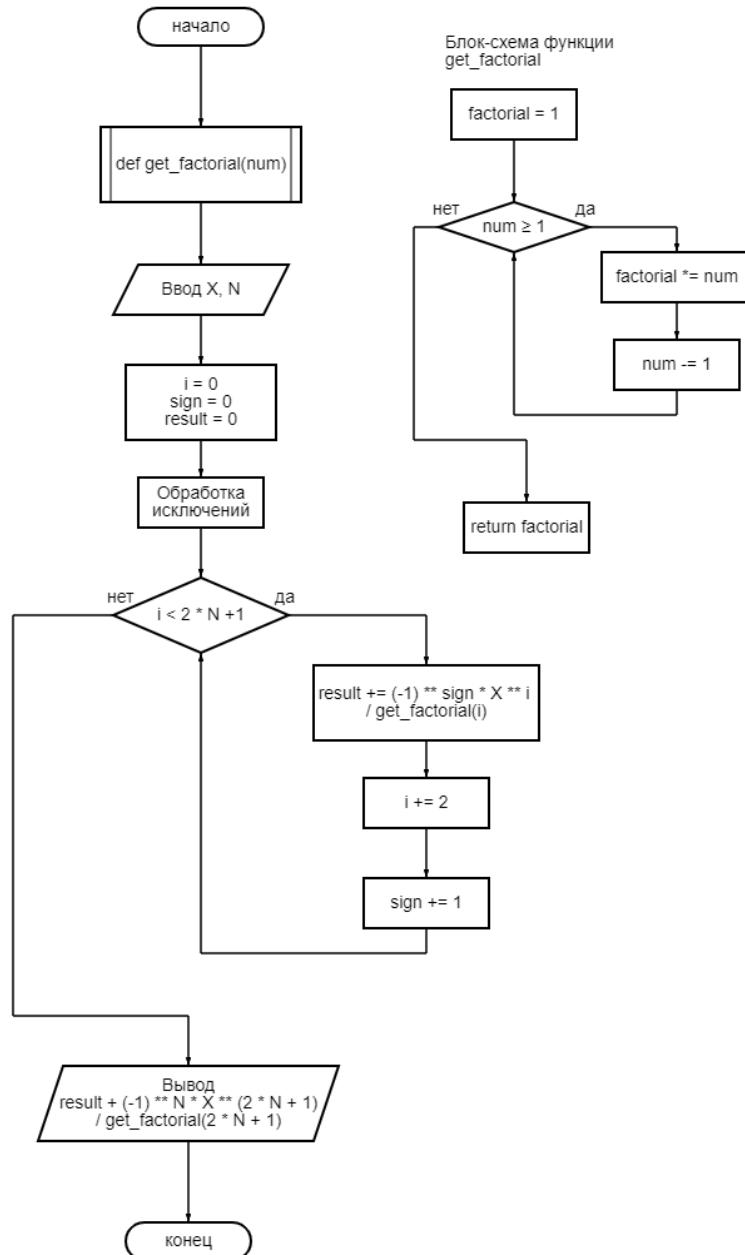
Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ циклической структуры в IDE PyCharm Community, первичные навыки работы с сервисом GitHub.

Постановка задачи №1:

Дано вещественное число X и целое число N (> 0). Найти значение выражения $X - X^3 / (3!) + X^5 / (5!) - \dots + (-1)^N * X^{2 * N+1} / ((2 * N+1)!)$ ($N! = 1 * 2 \dots N$). Полученное число является приближенным значением функции \sin в точке X.

Тип алгоритма: циклический

Блок схема алгоритма:



Текст программы:

```
# Дано вещественное число X и целое число N (> 0).
# Найти значение выражения X - X3/(3!) + X5/(5!) - ... + (-1) ** N * X
** (2 * N+1) /((2 * N+1) !) (N! = 12 ...N).
# Полученное число является приближенным значением функции sin в точке
X.

# Функция для нахождения факториала
def get_factorial(num):
    factorial = 1
    while num >= 1:
        factorial *= num
        num -= 1
    return factorial

X = input('Введите число: ')
N = input('Введите целое положительное число: ')
i = 1
sign = 0
result = 0

# Обработка исключений
while type(X) != float:
    try:
        X = float(X)
    except ValueError:
        print(f'Некорректный ввод - {X}!')
        X = input('Введите число: ')

while type(N) != int:
    try:
        N = int(N)
        if N <= 0:
            print(f'Некорректный ввод - {N}!')
            N = input('Введите целое положительное число: ')
    except ValueError:
        print(f'Некорректный ввод - {N}!')
        N = input('Введите целое положительное число: ')

# Нахождение значения выражения
while i < 2 * N + 1:
    result += (-1) ** sign * X ** i / get_factorial(i)
    i += 2
    sign += 1

print(f'Значение выражения равно {result + (-1) ** N * X ** (2 * N +
1) / get_factorial(2 * N + 1)}')
print('Программа успешно завершена!')
```

Протокол работы программы:

Введите число: 2

Введите целое положительное число: 10

Значение выражения равно 0.909297426825682

Программа успешно завершена!

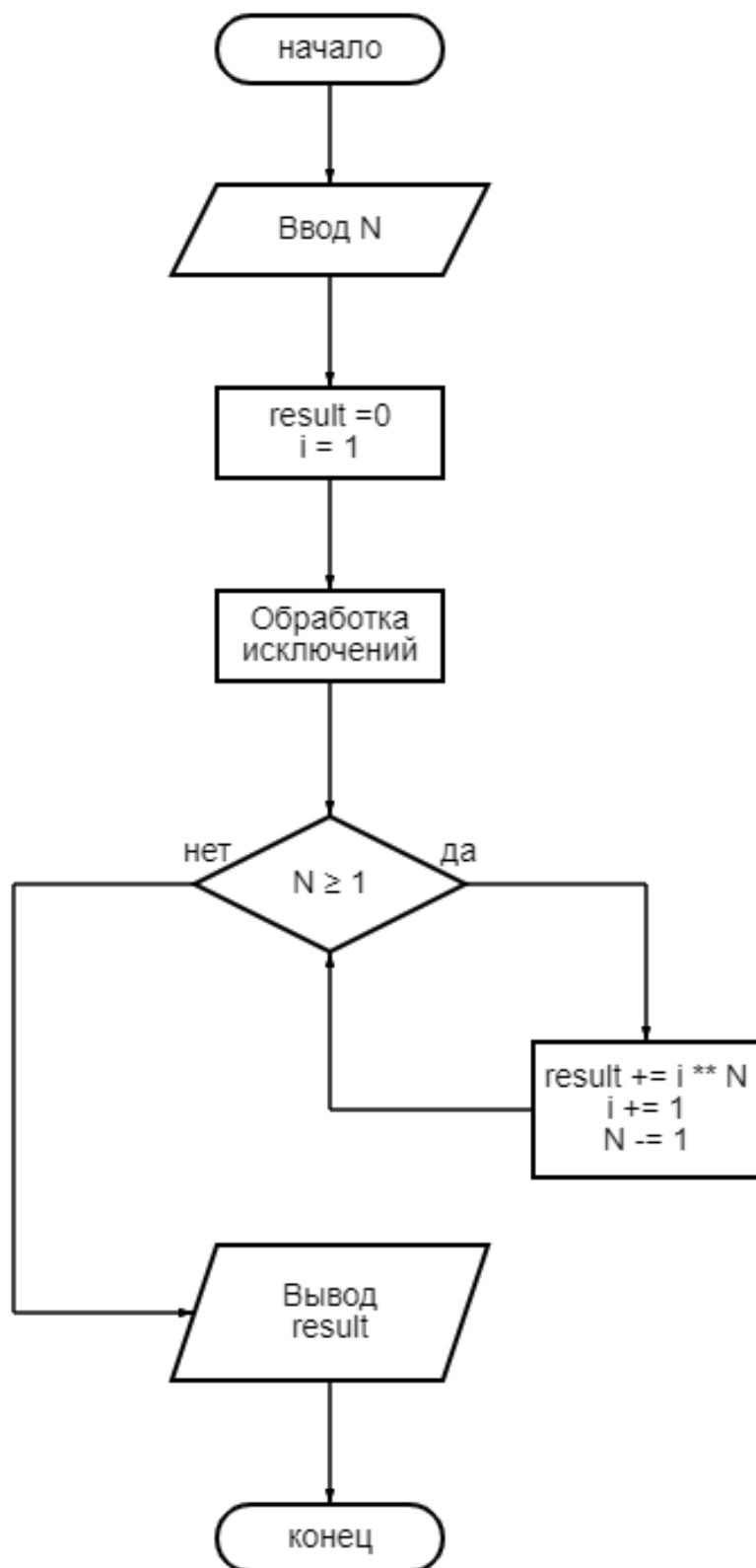
Process finished with exit code 0

Постановка задачи №2:

Дано целое число $N (> 0)$. Найти сумму $1^N + 2^{N-1} + \dots + N^1$.

Тип алгоритма: циклический

Блок схема алгоритма:



Текст программы:

```
# Дано целое число N (> 0). Найти сумму 1N + 2N-1 + ... + N1.

N = input('Введите целое положительное число: ')
result = 0
i = 1

# Обработка исключений
while type(N) != int:
    try:
        N = int(N)
        if N <= 0:
            print(f'Некорректный ввод - {N}!')
            N = input('Введите целое положительное число: ')
    except ValueError:
        print(f'Некорректный ввод - {N}!')
        N = input('Введите целое положительное число: ')

# Обработка чисел
while N >= 1:
    result += i ** N
    i += 1
    N -= 1

print(f'Сумма равна {result}\nПрограмма успешно завершена!')
```

Протокол работы программы:

Введите целое положительное число: 5

Сумма равна 65

Программа успешно завершена!

Process finished with exit code 0

Вывод: в процессе выполнения практического занятия выработал навыки составления программ циклической структуры в IDE PyCharm Community, приобрел первичные навыки работы с сервисом GitHub.

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

Готовые программные коды выложены на GitHub