

## Практическое занятие № 4

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

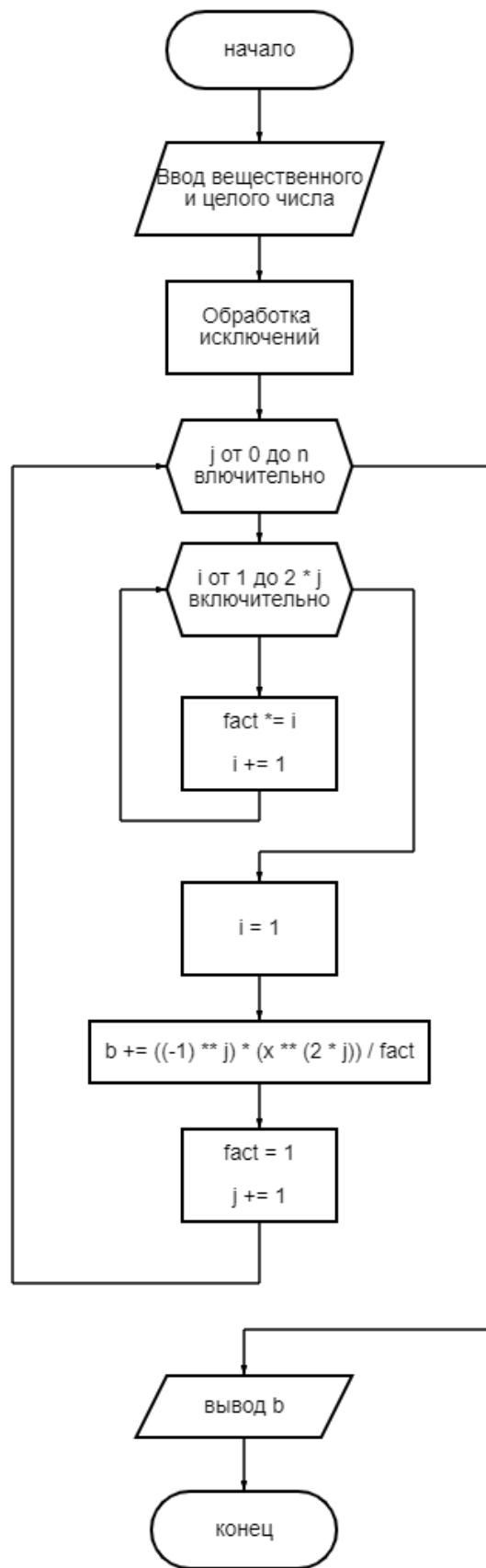
**Цели:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составление программ циклической структуры в IDE PyCharm Community, первичные навыки работы с сервисом GitHub.

### Постановка задачи 1.

Дано вещественное число X и целое число N ( $> 0$ ). Найти значение выражения  $1 - X^2/(2!) + X^4/(4!) - \dots + (-1)^N * X^{2^N}/(2^N N!)$  ( $N! = 12 \dots N$ ). Полученное число является приближенным значением функции cos в точке X.

**Тип алгоритма:** Циклический

## Блок-схема алгоритма:



## Текст программы:

```
x = input('Введите вещественное число: ')
n = input('Введите целое число: ')

while type(x) != float:           # обработка исключений
    try:
        x = float(x)
    except ValueError:
        print('Неверно ввели вещественное число!')
        x = input('Попробуйте еще раз: ')

while type(n) != int:
    try:
        n = int(n)
        if not n > 0:
            print('Число отрицательное!')
            n = input('Попробуйте еще раз: ')
    except ValueError:
        print('Неверно ввели целое число!')
        n = input('Попробуйте еще раз: ')

b = 0
fact = 1
j = 0
i = 1
while j <= n:
    while i <= 2 * j:
        fact *= i           # вычисление факториала
        i += 1
    i = 1
    b += ((-1) ** j) * (x ** (2 * j)) / fact      # вычисление
    fact = 1
    j += 1
print(f'косинус числа {x} равен {b} с точностью {n} знаков.')
```

## Протокол работы программы:

Введите вещественное число: 5.67

Введите целое число: 10

косинус числа 5.67 равен 0.8178509659237769 с точностью 10 знаков.

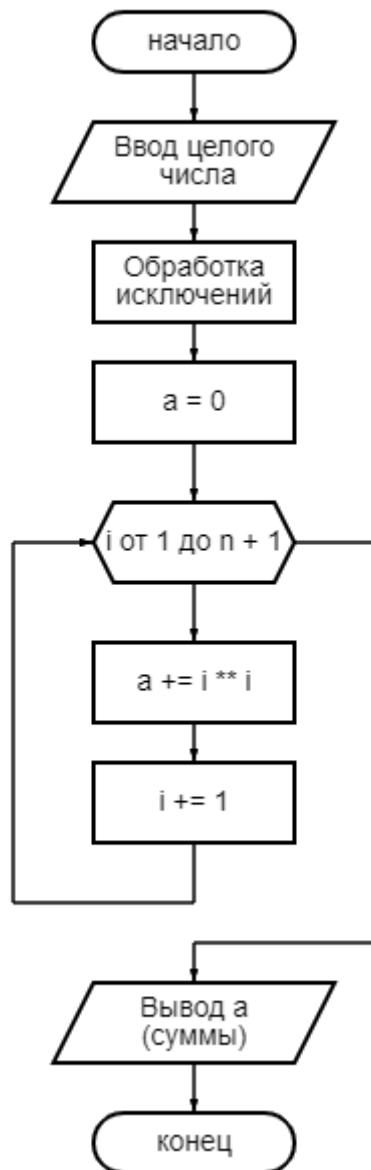
Process finished with exit code 0

## Постановка задачи 2.

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

**Тип алгоритма:** Циклический

**Блок-схема алгоритма:**



## Текст программы:

```
n = input('Введите целое число: ')
while type(n) != int:          # обработка исключений
    try:
        n = int(n)
        if not n > 0:
            print('Число отрицательное!')
            n = input('Попробуйте еще раз: ')
    except ValueError:
        print('Неправильно ввели целое число!')
        n = input('Попробуйте еще раз: ')
a = 0
i = 1
while i < n+1:
    a += i ** i
    i += 1
print(f'Сумма 1 ** 1 + 2 ** 2 + ... + N ** N равно: {a}')
```

## Протокол работы программы:

Введите целое число: 5

Сумма 1 \*\* 1 + 2 \*\* 2 + ... + N \*\* N равно: 3413

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрёл навыки составление программ циклической структуры в IDE PyCharm Community, первичные навыки работы с сервисом GitHub. Был использован цикл while.

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

Готовые программные коды выложены на GitHub.