

Практическое занятие № 4

Тема: составление программ циклической структуры в IDE PyCharm Community.

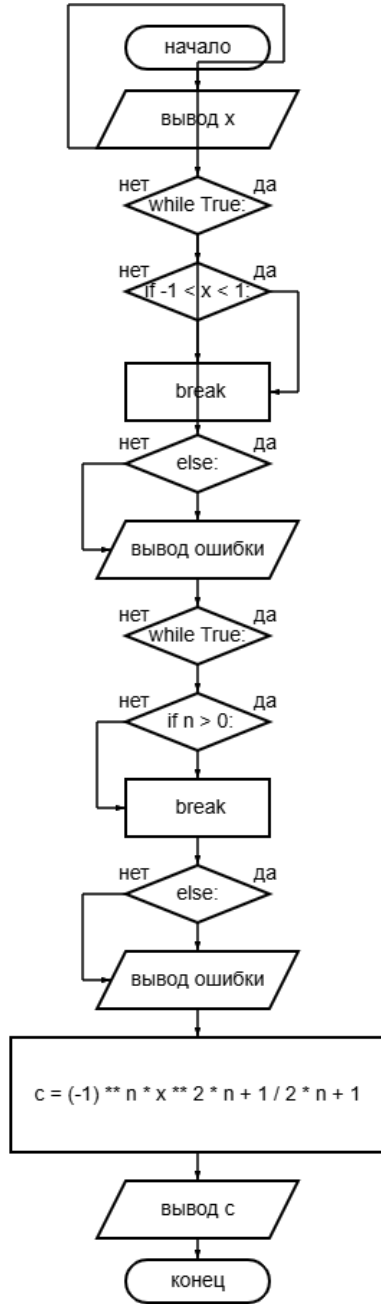
Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ циклической структуры в IDE PyCharm Community.

Постановка задачи:

1. Дано вещественное число X ($|X| < 0$). Найти значение выражения $X - X^3/3 + X^5/5 - \dots + (-1)^N X^{2N+1}/(2N+1)$. Полученное число является приближенным значением функции \arctg в точке X .

Тип алгоритма: циклический

Блок-схема алгоритма:



Текст программы:

```
# Дано вещественное число X ( $|X| < 1$ ) и целое число N ( $> 0$ ). Найти значение
# выражения
#  $X - X^3$ 
#  $\frac{1}{3} + X^5$ 
#  $\frac{1}{5} - \dots + (-1)^N X$ 
#  $2N + 1 / (2N + 1)$ . Полученное число является приближенным
# значением функции arctg в точке X.

while True:
    try:
        x = float(input("Введите вещественное число в диапазоне (-1, 1): "))
        if -1 < x < 1:
            break # Выходим из цикла, если x ввели правильно
        else:
            print("Это ведь не вещественное число!!!")
    except ValueError:
        print("Вы ввели текст, а надо вещественное число!")

while True:
    try:
        n = int(input("Введите положительное целое число: "))
        if n > 0:
            break # Выходим из цикла, если n ввели правильно
        else:
            print("Число должно быть положительным!")
    except ValueError:
        print("Вы ввели текст, а надо число!")

c = (-1) ** n * x ** 2 * n + 1 / 2 * n + 1
print(c)
```

Протокол работы программы:

```
C:\Users\user\AppData\Local\Programs\Python\Python312\py
Введите вещественное число в диапазоне (-1, 1): 0.45
Введите положительное целое число: 34
24.885

Process finished with exit code 0
```

Вывод: в процессе выполнения практического занятия выработал(а) навыки в составлении программ ветвящейся структуры в IDE PyCharm Community. Были использованы языковые конструкции try, except, break, if, else.

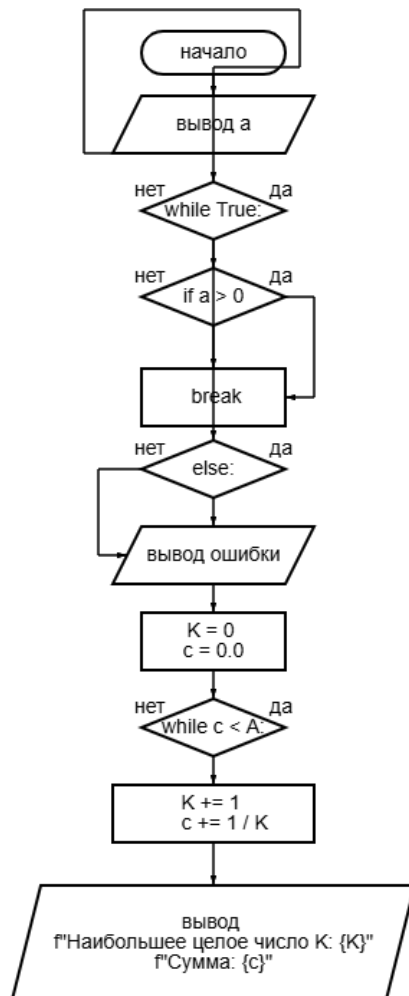
Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.

Постановка задачи:

2. Дано число A (>1). Вывести наибольшее из целых чисел K , для которых сумма $1 + 1/2 + \dots + 1/K$ будет меньше A , и саму эту сумму.

Тип алгоритма: циклический

Блок-схема алгоритма:



Текст программы:

```
A = input("Введите число A > 1: ")
while True:
    try:
        A = float(A)
        if A > 1:
            break
        else:
            print("A должно быть больше 1")
    except ValueError:
        print("Вы ввели текст, а надо число!")
        break
K = 0
c = 0.0

while c < A:
    K += 1
    c += 1 / K

K -= 1
c -= 1 / (K + 1)

print(f"Наибольшее целое число K: {K}")
print(f"Сумма: {c}")
```

Протокол работы программы:

```
C:\Users\user\AppData\Local\Programs\Python\Python312\
Введите число A > 1: 3
Наибольшее целое число K: 10
Сумма: 2.9289682539682538

Process finished with exit code 0
```

Вывод: в процессе выполнения практического занятия выработал(а) навыки в составлении программ ветвящейся структуры в IDE PyCharm Community. Были использованы языковые конструкции try, except, break, if, else.

Выполнены разработка кода, отладка, тестирование, оптимизация программного кода. Готовые программные коды выложены на GitHub.