

## Практическое занятие № 4

**Тема:** Составление программ с функциями в IDE PyCharm Community.

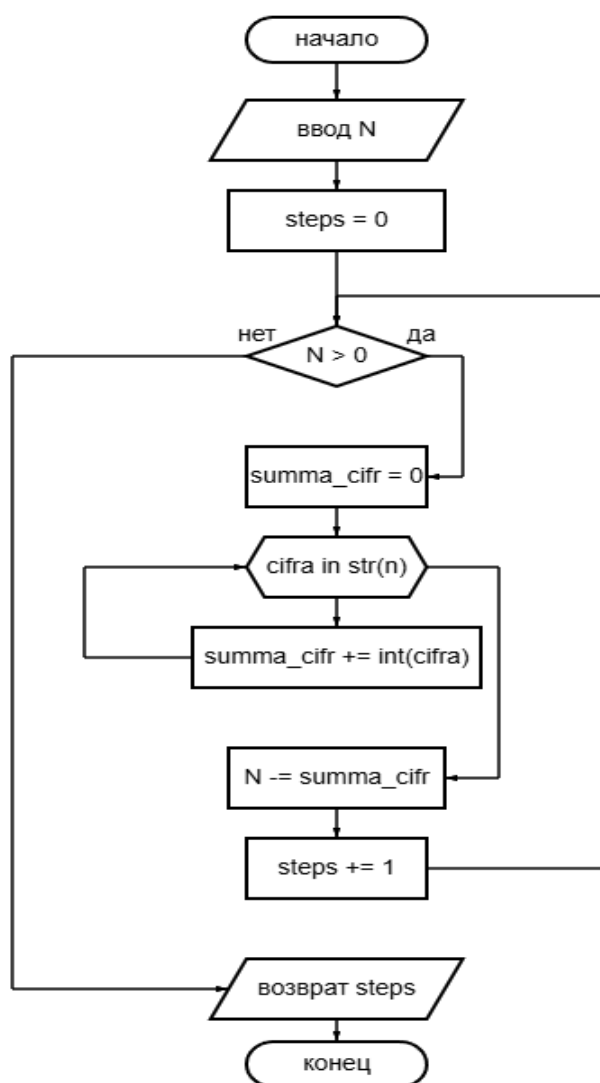
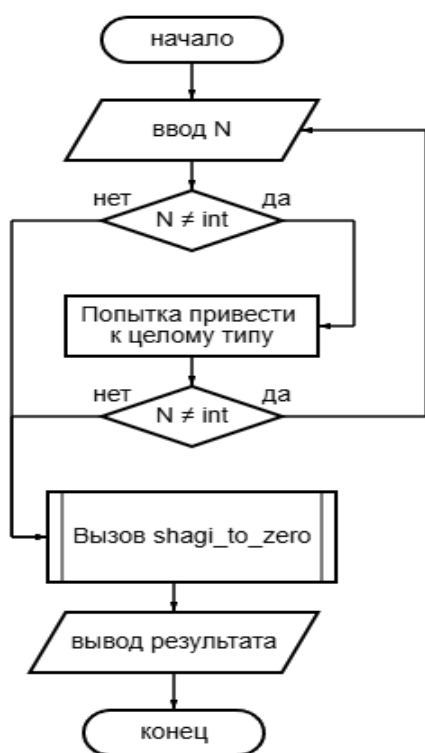
**Цель:** закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ с функциями в IDE PyCharm Community.

### Постановка задачи №1

Разработать программу, находящую ответ на задачу: составить функцию решения задачи: из заданного числа вычли сумму его цифр. Из результата вновь вычли сумму его цифр и т. д. Через сколько таких действий получится ноль?

**Тип алгоритма:** циклический.

**Блок-схема алгоритма:**



### Текст программы:

```
# Из заданного числа вычли сумму его цифр.
# Из результата вновь вычли сумму его цифр и т. д.
# Через сколько таких действий получится нуль?

def shagi_to_zero(n):
    steps = 0
    while n > 0:
        summa_cifr = 0          # Вычисляем сумму цифр
        for cifra in str(n):    # Проходим по каждой цифре числа
            summa_cifr += int(cifra) # Превращаем цифру в число и добавляем к
сумме
        n -= summa_cifr         # Вычитаем сумму цифр из числа
        steps += 1              # Увеличиваем счетчик шагов
    return steps
N = input("Введите число N: ") # Ввод N
while type(N) != int:          # обработка исключений для N
    try:
        N = int(N)
    except ValueError:
        print("Неправильно ввели!")
        N = input("Введи N заново: ")
result = shagi_to_zero(N)
print("Количество шагов, необходимых для достижения нуля:", result) # Выводим
результат
```

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

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

### Протокол работы программы:

Введите число N: 37

Количество шагов, необходимых для достижения нуля: 4

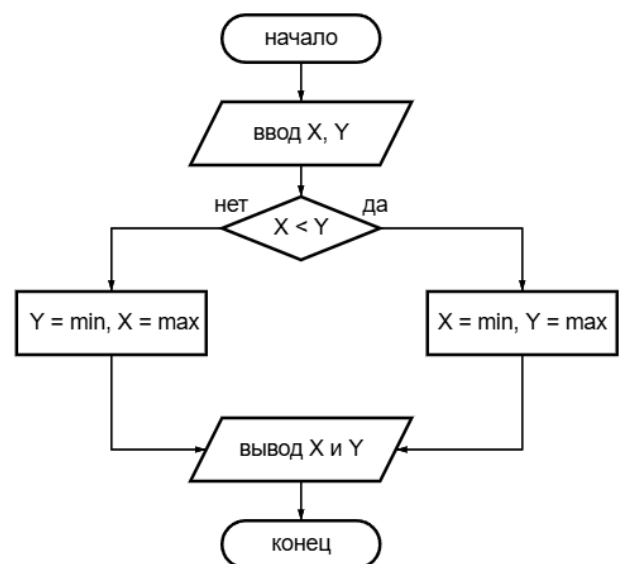
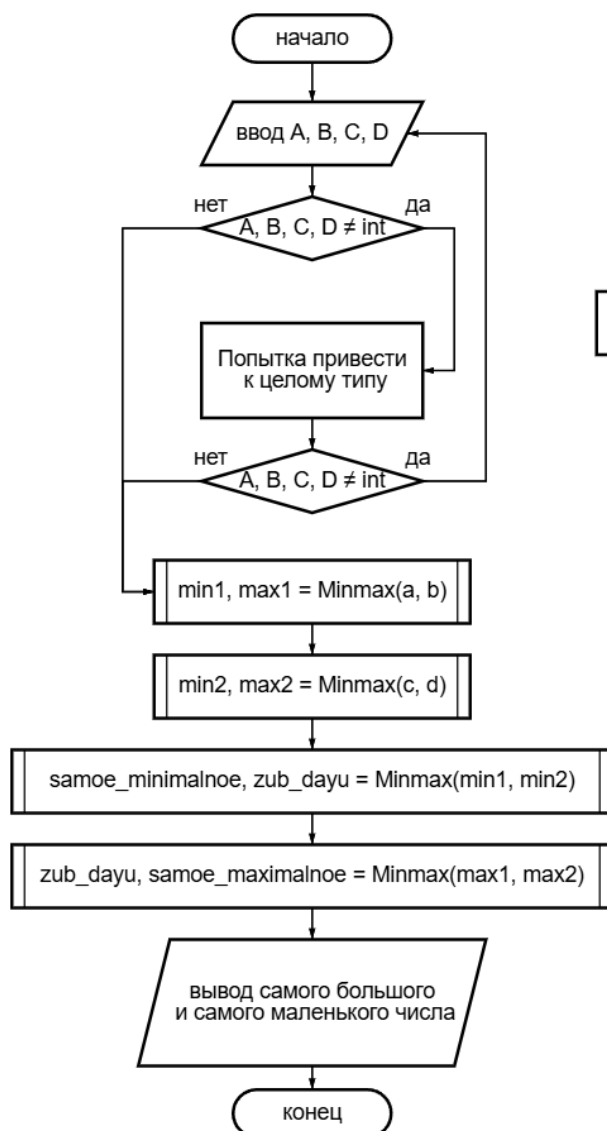
Process finished with exit code 0

## Постановка задачи №2

Разработать программу, находящую наибольшее целое число  $K$ , при котором выполняется неравенство  $3^K < N$ .

**Тип алгоритма:** циклический.

**Блок-схема алгоритма:**



## Текст программы:

```
# Используя четыре вызова функции Minmax, найти минимальное и максимальное из
данных чисел A, B, C, D.
a = input('Введи число A: ')
while type(a) != int:
    try:
        a = int(a)
    except ValueError:
        print('Неправильно ввели!')
        a = input("Введи A заново: ")
b = input('Введи число B: ')
while type(b) != int:
    try:
        b = int(b)
    except ValueError:
        print('Неправильно ввели!')
        b = input("Введи B заново: ")
c = input('Введи число C: ')
while type(c) != int:
    try:
        c = int(c)
    except ValueError:
        print('Неправильно ввели!')
        c = input("Введи C заново: ")
d = input('Введи число D: ')
while type(d) != int:
    try:
        d = int(d)
    except ValueError:
        print('Неправильно ввели!')
        d = input("Введи D заново: ")

def Minmax(X, Y):
    if X < Y:
        return X, Y
    else:
        return Y, X

min1, max1 = Minmax(a, b) # Минимум и максимум для A и B
min2, max2 = Minmax(c, d) # Минимум и максимум для C и D
самое_minimalное, зуб_дагу = Minmax(min1, min2) # Минимум из двух минимальных
значений
зуб_дагу, самое_maximalное = Minmax(max1, max2) # Максимум из двух максимальных
значений

print('Минимальное число: ', самое_minimalное)
print('Максимальное число: ', самое_maximalное)
```

### **Протокол работы программы:**

Введи число A: 56

Введи число B: 67

Введи число C: 78

Введи число D: 89

Минимальное число: 56

Максимальное число: 89

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал(а) навыки составления программ с функциями в IDE PyCharm Community.

Были использованы языковые конструкции def, while, try, except, return.

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

Готовые программные коды выложены на GitHub.