

### Практическое занятие № 3

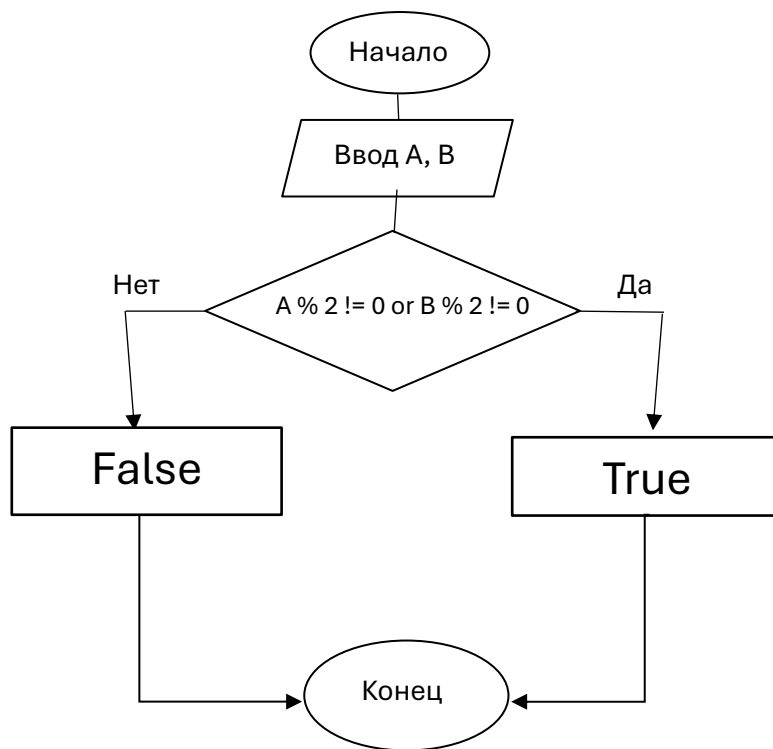
**Тема:** Составление программ ветвящейся структуры в IDE PyCharm Community.

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

**Постановка задачи:** Разработать программу проверяющую истинность высказывания: «Хотя бы одно из чисел А и В нечетное».

**Тип алгоритма:** Ветвящийся.

**Блок-схема алгоритма:**



### Текст программы:

```
'''
Вар. 9
Даны два целых числа: А, В.
Проверить истинность высказывания: «Хотя бы одно из чисел А и В нечетное».
'''

A = input("Введите первое целое число: ")
while type(A) != int: #проверка на целое число
    try:
        A = int(A)
    except:
        print("Ошибка!")
        A = input ("Повторите попытку: ")

B = input("Введите второе целое число: ")
while type(B) != int:
    try:
        B = int(B)
    except:
        print("Ошибка!")
        B = input("Повторите попытку: ")

if A % 2 != 0 or B % 2 != 0:
    print("True")
else:
    print("False")
```

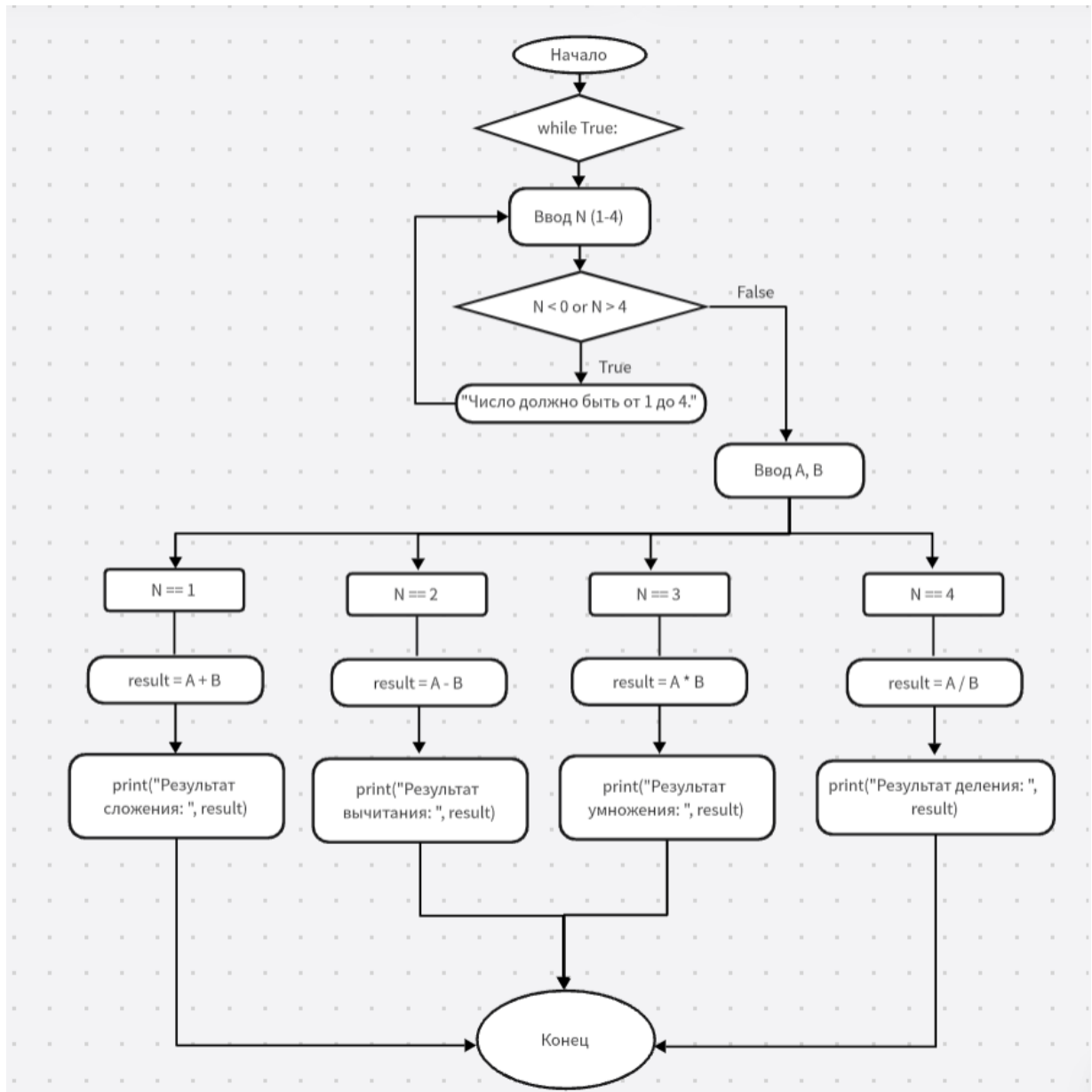
### Протокол работы программы:

Введите первое целое число: 12.2  
Ошибка!  
Повторите попытку: 12  
Введите второе целое число: 5.5  
Ошибка!  
Повторите попытку: 5  
True

**Постановка задачи:** Арифметические действия над числами пронумерованы следующим образом: 1 — сложение, 2 — вычитание, 3 — умножение, 4 — деление. Дан номер действия N (целое число в диапазоне 1-4) и вещественные числа А и В (В не равно 0). Выполнить над числами указанное действие и вывести результат.

**Тип алгоритма:** Циклический.

Блок-схема алгоритма:



Текст программы:

```

'''
Вер. 9
Арифметические действия над числами пронумерованы следующим образом: 1 –
сложение, 2 – вычитание, 3 – умножение, 4 – деление. Дан номер действия N
(целое число в диапазоне 1-4) и вещественные числа A и B (B не равно 0).
Выполнить
над числами указанное действие и вывести результат.
'''

def sum(A, B):
    return A + B

def sub(A, B):
    return A - B

def mltp(A, B):
    return A * B

def div(A, B):
    return A / B

while True: #Проверка действия на целое число от 1 до 4
    N = input("Введите число от 1 до 4 (1 = сложение, 2 = вычитание, 3 =
умножение, 4 = деление): ")
    try:
        operation = int(N)
        if operation < 1 or operation > 4:
            print("Число должно быть от 1 до 4.")
            continue
        break
    except ValueError:
        print("Ошибка! Повторите попытку.")

A = input("Введите первое вещественное число A: ")
while type(A) != float: #проверка на вещественное число
    try:
        A = float(A)
    except:
        print("Ошибка!")
        A = input ("Повторите попытку: ")

B = input("Введите второе вещественное число B: ")
while type(B) != float: #проверка на вещественное число
    try:
        B = float(B)
    except:
        print("Ошибка!")
        B = input ("Повторите попытку: ")

if operation == 1:
    result = sum(A,B)
    print("Результат сложения: ", result)
elif operation == 2:
    result = sub(A,B)
    print("Результат вычитания: ", result)
elif operation == 3:
    result = mltp(A,B)
    print("Результат умножения: ", result)
elif operation == 4:
    if B == 0:
        print("Ошибка: деление на ноль!")
    else:

```

```
result = div(A,B)
print("Результат деления:", result)
```

### **Протокол работы программы:**

Введите число от 1 до 4 (1 = сложение, 2 = вычитание, 3 = умножение, 4 = деление): 5

Число должно быть от 1 до 4.

Введите число от 1 до 4 (1 = сложение, 2 = вычитание, 3 = умножение, 4 = деление): 1

Введите первое вещественное число A: 12.5

Введите второе вещественное число B: 4.55

Результат сложения: 17.05

**Вывод:** Закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ ветвящейся структуры в IDE PyCharm Community, составил программу ветвящейся и циклической структуры. Готовый программный код выложен на GitHub.