

### 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.

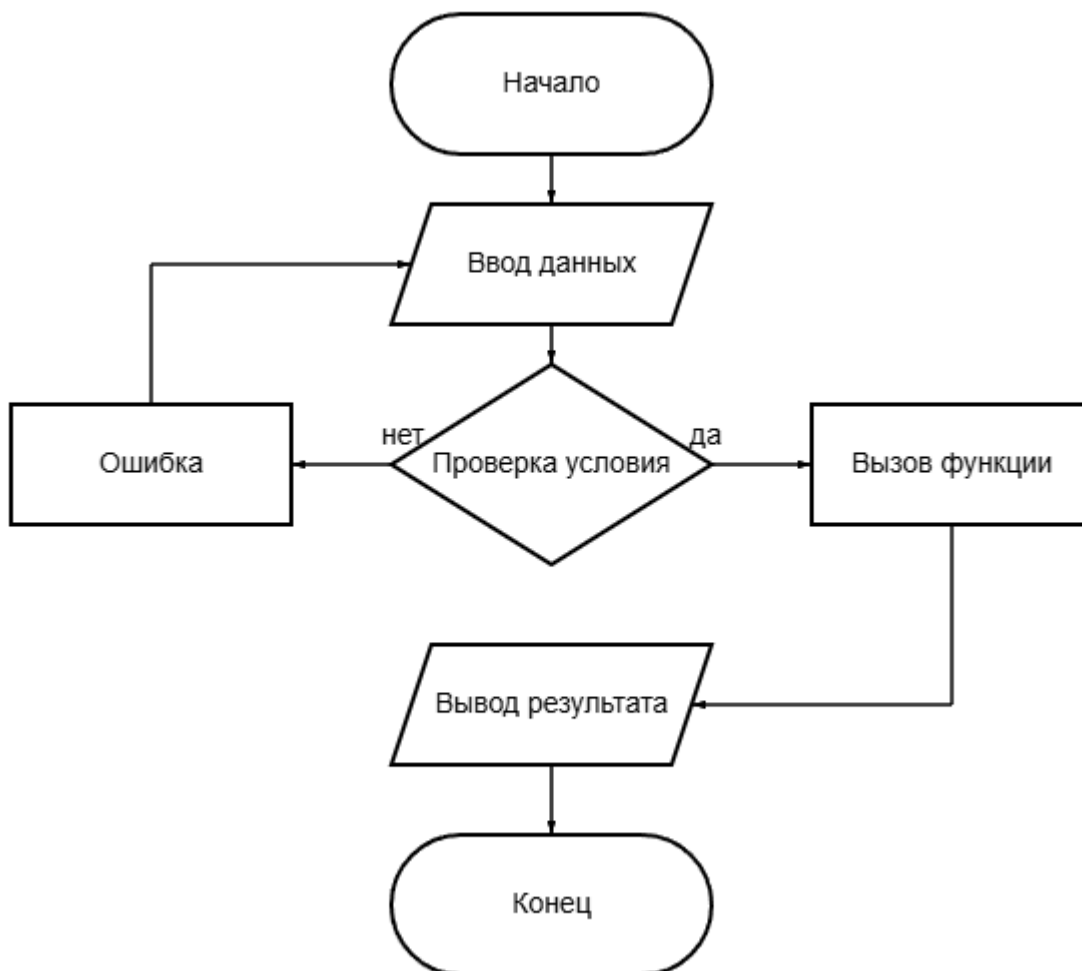
#### Первая работа

##### Постановка задачи.

Необходимо проверить истинность высказывания: «Точка с координатами  $(x, y)$  лежит внутри прямоугольника, левая верхняя вершина которого имеет координаты  $(x_1, y_1)$ , правая нижняя —  $(x_2, y_2)$ , а стороны параллельны координатным осям».

**Тип алгоритма:** ветвистый.

**Блок-схема алгоритма:**



### Текст программы:

```
#Даны числа x, y, x1, y1, x2, y2.
#Проверить истинность высказывания:
#«Точка с координатами (x, y) лежит внутри прямоугольника,
#левая верхняя вершина которого имеет координаты (x1, y1),
#правая нижняя — (x2, y2), а стороны параллельны координатным осям».
def coordinates_rectangle(x, y, x1, y1, x2, y2):
    if x1 >= x2 or y1 >= y2:
        raise ValueError("Координаты должны быть корректными: x1 < x2 и y1 < y2")
    return (x1 < x < x2) and (y1 < y < y2)
while True:
    try:
        x = int(input("Введите координату x: "))
        x1 = int(input("Введите координату x1: "))
        x2 = int(input("Введите координату x2: "))
        y = int(input("Введите координату y: "))
        y1 = int(input("Введите координату y1: "))
        y2 = int(input("Введите координату y2: "))
        result = coordinates_rectangle(x, y, x1, y1, x2, y2)
        print(result)
        break
    except ValueError as e:
        print(f"Ошибка: {e}. Попробуйте снова.")
```

### Протокол работы программы:

True

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ ветвистой структуры в IDE PyCharm Community. Были использованы языковые конструкции: def, return, if, try, except, raise, while.

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

Готовые программные коды выложены на GitHub.

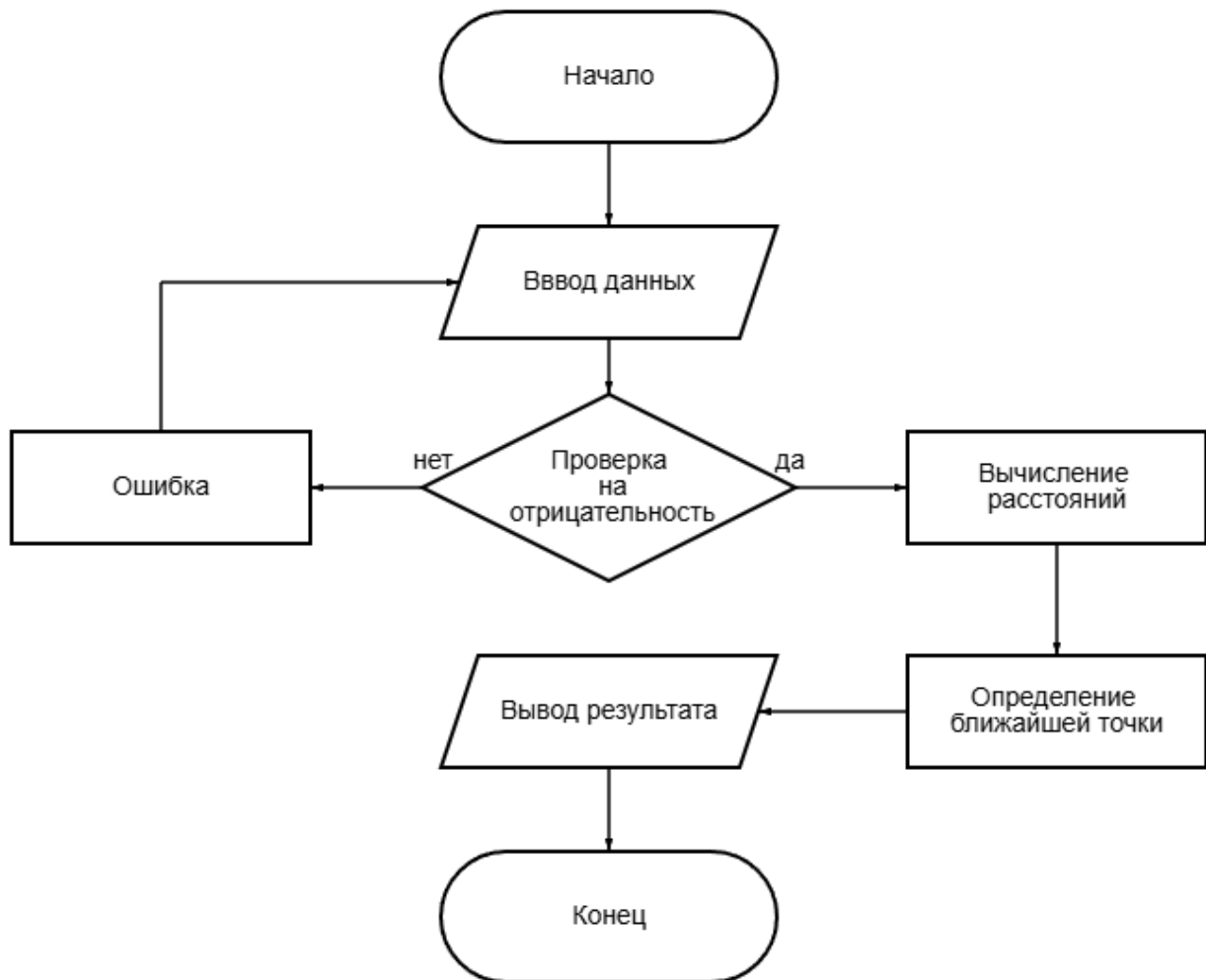
## Вторая работа

### Постановка задачи.

Необходимо написать функцию, которая определяет, какая из двух заданных точек (В или С) находится ближе к третьей точке (А) на числовой оси. Функция также должна вычислять расстояние от точки А до ближайшей точки.

**Тип алгоритма:** ветвистый.

**Блок-схема алгоритма:**



## Текст программы:

```
#На числовой оси расположены три точки: A, B, C.
#Определить, какая из двух последних точек (B или C) расположена ближе к A,
#и вывести эту точку и ее расстояние от точки A.

def numeric_axis(A, B, C):
    A = float(A)
    B = float(B)
    C = float(C)
    if A < 0 or B < 0 or C < 0:
        raise ValueError("Числа не должны быть отрицательными")
    if A > B:
        distance_B = A - B
    else:
        distance_B = B - A
    if A > C:
        distance_C = A - C
    else:
        distance_C = C - A
    if distance_B < distance_C:
        axis_point = "B"
        main_distance = distance_B
    else:
        axis_point = "C"
        main_distance = distance_C
    return axis_point, main_distance

while True:
    try:
        A = int(input("Введите точку A: "))
        B = int(input("Введите точку B: "))
        C = int(input("Введите точку C: "))
        axis_result = numeric_axis(A, B, C)
        print(f"Самая близкая точка к A: {axis_result[0]}")
        print(f"Расстояние от точки A: {axis_result[1]}")
        break
    except ValueError as e:
        print(f"Ошибка: {e}. Попробуйте снова.")
```

**Протокол работы программы:**

Самая близкая точка к A: C

Расстояние от точки A: 1

Process finished with exit code 0

**Вывод:** в процессе выполнения практического занятия выработал навыки составления программ ветвистой структуры в IDE PyCharm Community. Были использованы языковые конструкции: def, return, if, f-строки, try, except, while, raise.

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

Готовые программные коды выложены на GitHub.