

Практическое занятие № 3

Тема: Составление программ ветвящейся структуры в IDE PyCharm Community.

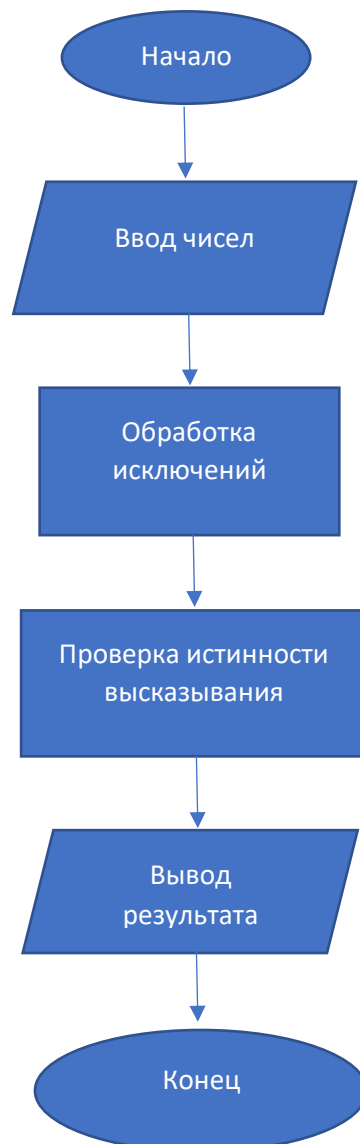
Цель: закрепить усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрести навыки составления программ ветвящейся структуры в IDE PyCharm Community.

Постановка задачи: Вариант 29.

1. Даны числа x, y, x_1, y_1, x_2, y_2 . 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: линейный

Блок-схема алгоритма:



```
1  # Вариант 29.
2  # 1. Даны числа x, y, x1, y1, x2, y2. Проверить истинность высказывания: "Точка с
3  # координатами (x, y) лежит внутри прямоугольника, левая вершина которого
4  # имеет координаты (x1, y1), правая нижняя - (x2, y2), а стороны параллельны
5  # координатным осям".
6
7  x = input('Введите целое число!') # Ввод числа
8  y = input('Введите целое число!')
9  x1 = input('Введите целое число!')
10 y1 = input('Введите целое число!')
11 x2 = input('Введите целое число!')
12 y2 = input('Введите целое число!')
13 while type(x) != int: # Проверка на целочисленность
14     try:
15         x = int(x)
16     except ValueError:
17         print('Введите целое число!')
18         x = input()
19 while type(y) != int:
20     try:
21         y = int(y)
22     except ValueError:
23         print('Введите целое число!')
24         y = input()
25 while type(x1) != int:
26     try:
27         x1 = int(x1)
28     except ValueError:
29         print('Введите целое число!')
30         x1 = input()
31 while type(y1) != int:
32     try:
33         y1 = int(y1)
34     except ValueError:
35         print('Введите целое число!')
36         y1 = input()
```

```
37 while type(x2) != int:
38     try:
39         x2 = int(x2)
40     except ValueError:
41         print('Введите целое число!')
42         x2 = input()
43 while type(y2) != int:
44     try:
45         y2 = int(y2)
46     except ValueError:
47         print('Введите целое число!')
48         y2 = input()
49 if (x1 < x < x2) and (y1 > y > y2): # Проверка условия
50     print('Входит в прямоугольник') # Вывод результата
51 else:
52     print('Не входит в прямоугольник')
53
```

Протокол работы программы:

Вводим числа: 5 4 -7 6 7 -6

Вывод: Входит в прямоугольник

Process finished with exit code 0

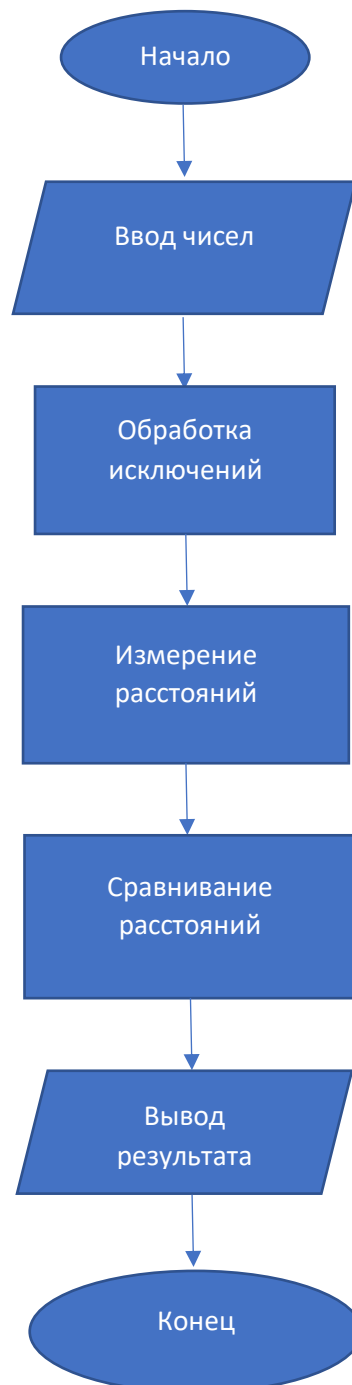
Постановка задачи: Вариант 29.

2. На числовой оси расположены три точки: А, В, С. Определить, какая из двух последних точек (В или С) расположена ближе к А, и вывести эту точку и ее расстояние от точки А.

Тип алгоритма: линейный

Блок-схема алгоритма:

Студент группы ИС-21 Черячукин К. А.



Студент группы ИС-21 Черячукин К. А.

```
1  # Вариант 29.
2  # 2. На числовой оси расположены три точки: A,B,C. Определить, какая из двух
3  # последних точек (В или С) расположена ближе к А, и вывести эту точку и ее
4  # расстояние от точки А.
5
6  a = input('Введите целое число!') # ввод переменных
7  b = input('Введите целое число!')
8  c = input('Введите целое число!')
9
10 while type(a) != int: # проверка на целочисленность
11     try:
12         a = int(a)
13     except ValueError:
14         print('Введите целое число')
15         a = input()
16
17 while type(b) != int:
18     try:
19         b = int(b)
20     except ValueError:
21         print('Введите целое число')
22         b = input()
23
24 while type(c) != int:
25     try:
26         c = int(c)
27     except ValueError:
28         print('Введите целое число')
29         c = input()
30
31 ac = abs(a - c) # расчеты
32 ab = abs(a - b)
33
34 if ac < ab: # проверки условий
35     print('a ближе')
36 if ab < ac:
37     print('b ближе')
38 if ab == ac:
39     print('расстояние одинаково')
```

Протокол работы программы:

Вводим числа: 4 3 2

Вывод: b ближе

Process finished with exit code 0

Вывод: закрепил усвоенные знания, понятия, алгоритмы, основные принципы составления программ, приобрел навыки составления программ ветвящейся структуры в IDE PyCharm Community.

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