Лабораторная работа №*2*

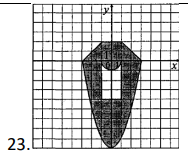
«***Разветвляющиеся алгоритмы***»

|  |  |
| --- | --- |
|  | Выполнил:  Студент группы *BC-209*  *Чураков Константин Эдуардович*  Вариант №*23*  Проверила:  Каргаполова Ю.А. |

Задание:

1) Шахматный конь за один ход может переместиться с одного заданного поля на другое (каждое поле задано двумя координатами— целыми числами от 1 до 8)

2) Надиться ли данная точка в этой фигуре



Код программы:

**from math import \***

**# Вариант 23**

**# Задание 1**

**sx = int(input('Введите x координату стартового поля (от 1 до 8): '))**

**sy = int(input('Введите y координату стартового поля (от 1 до 8): '))**

**ex = int(input('Введите x координату конечного поля (от 1 до 8): '))**

**ey = int(input('Введите y координату конечного поля (от 1 до 8): '))**

**xfabs = abs(sx - ex)**

**yfabs = abs(sy - ey)**

**fabs = (xfabs == 2 and yfabs == 1) or (xfabs == 1 and yfabs == 2)**

**start = sx >= 1 and sx <= 8 and sy >= 1 and sy <= 8**

**end = ex >= 1 and ex <= 8 and ey >= 1 and ey <= 8**

**if start and end and fabs:**

**print('Конь может сходить на это поле')**

**else:**

**print('Конь не может сходить на это поле')**

**print()**

**# Задание 2**

**px = float(input('Введите x координату: '))**

**py = float(input('Введите y координату: '))**

**if py >= 0:**

**shape1 = px <= 1 and px >= -1 and py <= 2 and py >= 0**

**# Алгоритм collision detection для треугольника и точки взят отсюда**

**# http://jeffreythompson.org/collision-detection/tri-point.php**

**x1 = -3**

**y1 = 0**

**x2 = -1**

**y2 = 2**

**x3 = -1**

**y3 = 0**

**areaOrig = abs((x2-x1)\*(y3-y1)-(x3-x1)\*(y2-y1))**

**area1 = abs((x1-px)\*(y2-py)-(x2-px)\*(y1-py))**

**area2 = abs((x2-px)\*(y3-py)-(x3-px)\*(y2-py))**

**area3 = abs((x3-px)\*(y1-py)-(x1-px)\*(y3-py))**

**shape2 = area1 + area2 + area3 == areaOrig**

**x1 = 3**

**y1 = 0**

**x2 = 1**

**y2 = 2**

**x3 = 1**

**y3 = 0**

**areaOrig = abs((x2-x1)\*(y3-y1)-(x3-x1)\*(y2-y1))**

**area1 = abs((x1-px)\*(y2-py)-(x2-px)\*(y1-py))**

**area2 = abs((x2-px)\*(y3-py)-(x3-px)\*(y2-py))**

**area3 = abs((x3-px)\*(y1-py)-(x1-px)\*(y3-py))**

**shape3 = area1 + area2 + area3 == areaOrig**

**if shape1 or shape2 or shape3:**

**print('Точка находиться в отмеченной области')**

**else:**

**print('Точка не находиться в отмеченной области')**

**else:**

**func1 = py >= pow(px, 2) - 9**

**func2 = py >= pow(px, 2) - 1**

**shape1 = px < 1 and px > -1 and py < 0 and py > -4**

**if shape1:**

**if func2:**

**print('Точка находиться в отмеченной области')**

**else:**

**print('Точка не находиться в отмеченной области')**

**else:**

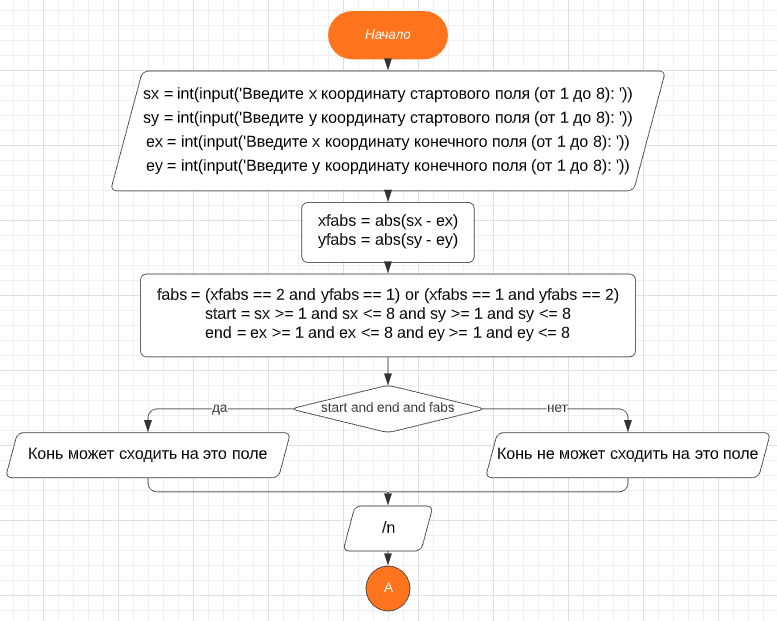
**if func1:**

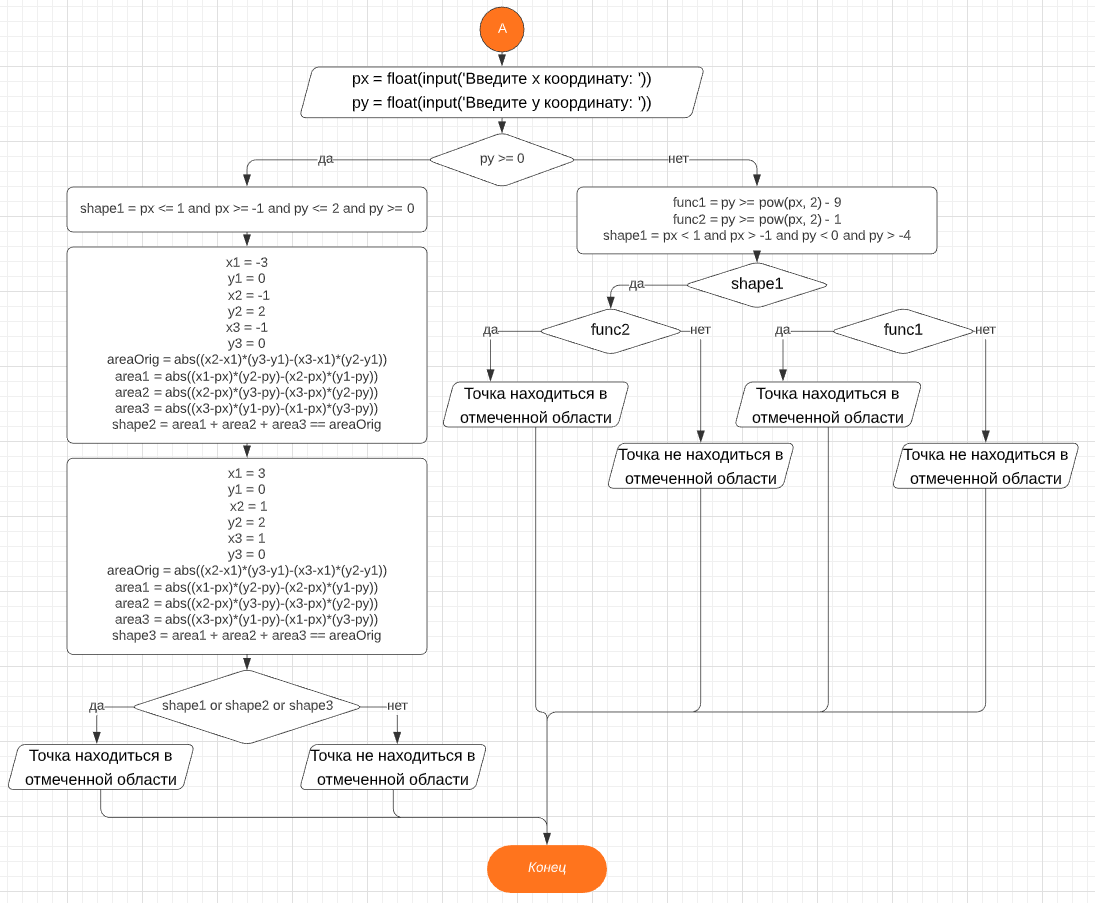
**print('Точка находиться в отмеченной области')**

**else:**

**print('Точка не находиться в отмеченной области')**

Блок схема к написанной программе:

**



Скриншоты компиляции написанной программы:

