**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**

**федеральное государственное автономное образовательное**

**учреждение высшего образования**

**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**

**(РУТ(МИИТ)**

Кафедра «Вычислительные системы, сети и информационная безопасность»

**Отчет По Дисциплине**

**«Программирование»**

**Лабораторная работа №1**

Направление: 09.03.01Информатика и вычислительная техника

Профиль:Вычислительные системы и сети

Выполнил:

студент группы УВВ-111

Ермаков Н.К.

Проверил:

ст. п. Цыганова Н.А.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(должность, ФИО)

Москва 2024 г.

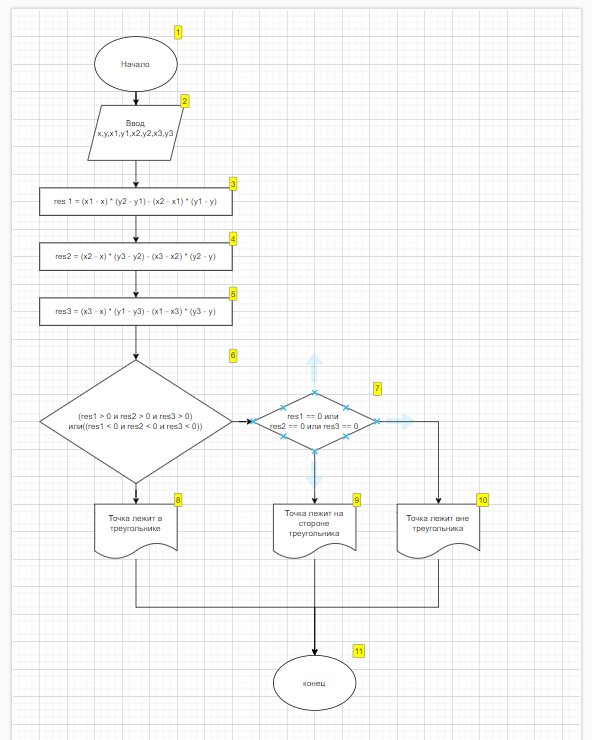
**Задание №1**

Задание: Заданы координаты точки x, y. Определить, принадлежит ли эта точка треугольнику с вершинами в точках (x1, y1), (x2, y2), (x3, y3).

**1 Таблица имён:**

|  |  |  |
| --- | --- | --- |
| Исходные данные | | |
| x | вещественный | -координата точки по оси х |
| y | вещественный | -координата точки по оси y |
| x1 | вещественный | -координата точки 1 треугольника по оси х |
| y1 | вещественный | -координата точки 1 треугольника по оси y |
| x2 | вещественный | -координата точки 2 треугольника по оси х |
| y2 | вещественный | -координата точки 2 треугольника по оси y |
| x3 | вещественный | -координата точки 3 треугольника по оси y |
| y3 | вещественный | -координата точки 3 треугольника по оси y |
| Рабочие переменные | | |
| res1 | вещественный | вычисляет ориентацию точки относительно 1 стороны треугольника. |
| res2 | вещественный | вычисляет ориентацию точки относительно 2  стороны треугольника. |
| res3 | вещественный | вычисляет ориентацию точки относительно 3 стороны треугольника. |

**2.** Блок схема:



**3** Отладочный пример:

Вариант 1:

1. Начало
2. Ввод с клавиатуры x, y, x1, y1, x2, y2, x3, y3
3. x = 1.0

y = 1.0

x1 = 0.0

y1 = 0.0

x2 = 1.0

y2 = 3.0

x3 = 3.0

y3 = 0.0

1. res1 = (x1 - x) \* (y2 - y1) - (x2 - x1) \* (y1 - y)

res1 = (0.0 - 1.0) \* (3.0 – 0.0) – (1.0 – 0.0) \* (0.0 – 1.0)

res1 = -2.0

res2 = (x2 - x) \* (y3 - y2) - (x3 - x2) \* (y2 - y)

res2 = (1.0 – 1.0) \* (0.0 – 3.0) – (3.0 – 1.0) \* (3.0 – 1.0)

res2 = -4.0

res3 = (x3 - x) \* (y1 - y3) - (x1 - x3) \* (y3 - y)

res3 = (0.0 – 1.0) \* (0.0 – 0.0) – (0.0 – 3.0) \* (0.0 – 1.0)

res3 = -3.0

1. проверка условий

(res1> 0 и res2> 0 и res3> 0) или (res1 <0 и res2 <0 и res3 <0)

1. Вывод на экран: Точка лежит в треугольнике
2. Конец

Вариант 2:

1. Начало
2. Ввод с клавиатуры: x, y, x1, y1, x2, y2, x3, y3
3. x = 1.0

y = 1.0

x1 = 0.0

y1 = 0.0

x2 = 1.0

y2 = 3.0

x3 = 3.0

y3 = Ноль

1. Вывод на экран: Введено некорректное значение
2. Конец

Вариант 3:

1. Начало
2. Ввод с клавиатуры: x, y, x1, y1, x2, y2, x3, y3
3. x = 0.5

y = 0.0

x1 = 0.0

y1 = 0.0

x2 = 1.0

y2 = 3.0

x3 = 3.0

y3 = 0.0

1. res1 = (x1 - x) \* (y2 - y1) - (x2 - x1) \* (y1 - y)

res1 = (0.0 – 0.5) \* (3.0 – 0.0) – (1.0 – 0.0) \* (0.0 – 0.0)

res1 = -1.5

res2 = (x2 - x) \* (y3 - y2) - (x3 - x2) \* (y2 - y)

res2 = (1.0 – 0.5) \* (0.0 – 3.0) – (3.0 – 1.0) \* (3.0 – 0.0)

res2 = -7.5

res3 = (x3 - x) \* (y1 - y3) - (x1 - x3) \* (y3 - y)

res3 = (0.0 – 0.5) \* (0.0 – 0.0) – (0.0 – 3.0) \* (0.0 – 0.0)

res3 = 0.0

1. проверка условий

(res1> 0 и res2> 0 и res3> 0) или (res1 <0 и res2 <0 и res3 <0)

res1 = 0 или res2 = 0 или res3 = 0

1. Вывод на экран: Точка лежит на стороне треугольника
2. Конец

Вариант 4:

1. Начало
2. Ввод с клавиатуры: x, y, x1, y1, x2, y2, x3, y3
3. x = 10.0

y = 10.0

x1 = 0.0

y1 = 0.0

x2 = 1.0

y2 = 3.0

x3 = 3.0

y3 = 0.0

1. res1 = (x1 - x) \* (y2 - y1) - (x2 - x1) \* (y1 - y)

res1 = (0.0 – 10.0) \* (3.0 – 0.0) – (1.0 – 0.0) \* (0.0 – 10.0)

res1 = -20.0

res2 = (x2 - x) \* (y3 - y2) - (x3 - x2) \* (y2 - y)

res2 = (1.0 – 10.0) \* (0.0 – 3.0) – (3.0 – 1.0) \* (3.0 – 10.0)

res2 = 41.0

res3 = (x3 - x) \* (y1 - y3) - (x1 - x3) \* (y3 - y)

res3 = (0.0 – 10.0) \* (0.0 – 0.0) – (0.0 – 3.0) \* (0.0 – 10.0)

res3 = -30.0

1. проверка условий

(res1> 0 и res2> 0 и res3> 0) или (res1 <0 и res2 <0 и res3 <0)

res1 = 0 или res2 = 0 или res3 = 0

1. Вывод на экран: Точка лежит вне треугольника
2. Конец

**4** Код программы

import java.util.Scanner; // импортирование библиотеки для ввода данных  
import java.util.InputMismatchException; // импортирование библиотеки для обработки искл. с клав.  
public class First {  
 public static void main(String[] args){  
 try{  
 Scanner in = new Scanner(System.in);// создание сканера  
 System.out.print("Введите через пробел x и y:");  
 float x = in.nextFloat();  
 float y = in.nextFloat();  
 System.out.print("Введите через пробел x1 и y1:");  
 float x1 = in.nextFloat();  
 float y1 = in.nextFloat();  
 System.out.print("Введите через пробел x2 и y2:");  
 float x2 = in.nextFloat();  
 float y2 = in.nextFloat();  
 System.out.print("Введите через пробел x3 и y3:");  
 float x3 = in.nextFloat();  
 float y3 = in.nextFloat();  
 float res1 = (x1 - x) \* (y2 - y1) - (x2 - x1) \* (y1 - y);// вычисляет ориентацию точки относительно прямой  
 float res2 = (x2 - x) \* (y3 - y2) - (x3 - x2) \* (y2 - y);// вычисляет ориентацию точки относительно прямой  
 float res3 = (x3 - x) \* (y1 - y3) - (x1 - x3) \* (y3 - y);// вычисляет ориентацию точки относительно прямой  
 if ((res1 > 0 & res2 > 0 & res3 > 0)||((res1 < 0 & res2 < 0 & res3 < 0))) {  
 System.out.println("Точка лежит в треугольнике");  
 } else if (res1 == 0 || res2 == 0 || res3 == 0) {  
 System.out.println("Точка лежит на стороне треугольника");  
 }  
 else {  
 System.out.println("Точка лежит вне треугольника");  
 }  
 }  
 catch(InputMismatchException x){  
 System.out.println("Введено некорректное значение");  
 }  
 }  
}

**5** Результат выполнения программы

1. В составленную программу вводятся данные, после чего в конце ее работы можно увидеть вывод одного из условий

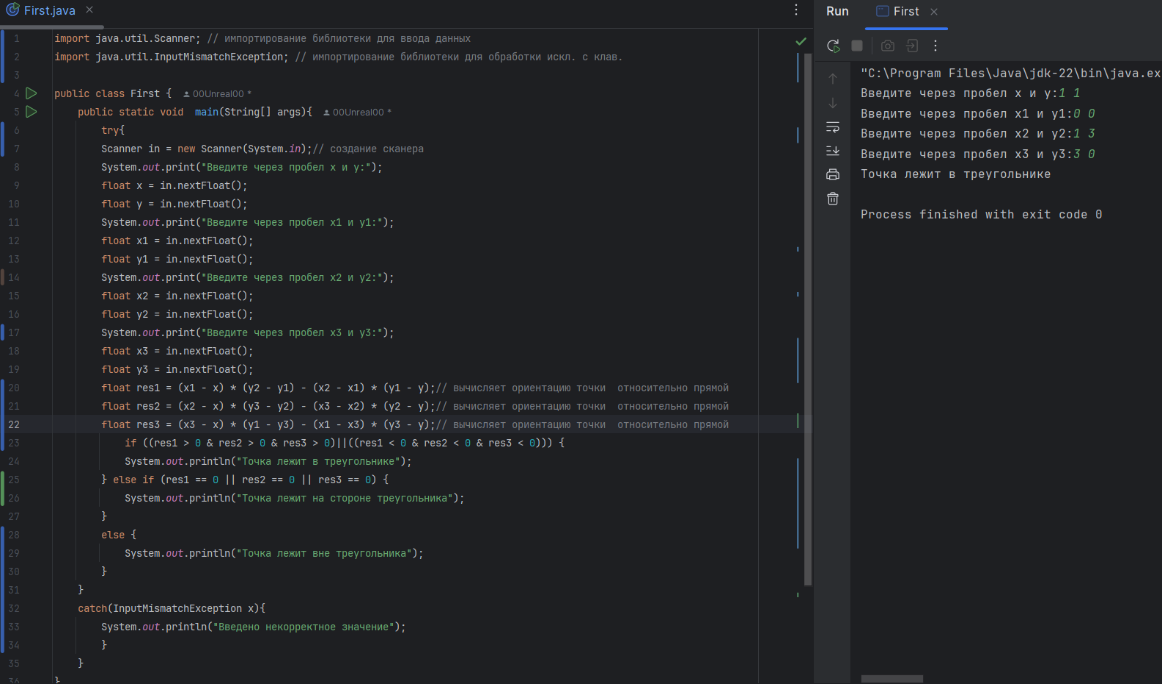


Рисунок 1 – Результат работы программы

1. В составленную программу вводятся данные, после чего в конце ее работы можно увидеть вывод одного из условий

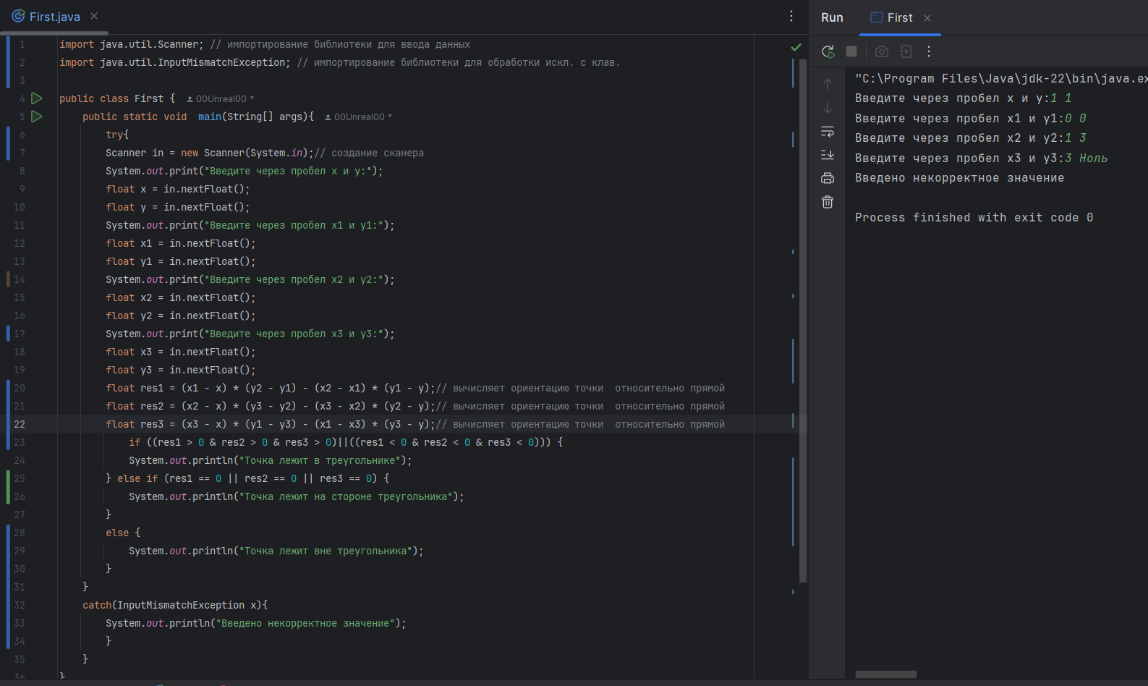


Рисунок 2 – Результат работы программы

1. В составленную программу вводятся данные, после чего в конце ее работы можно увидеть вывод одного из условий

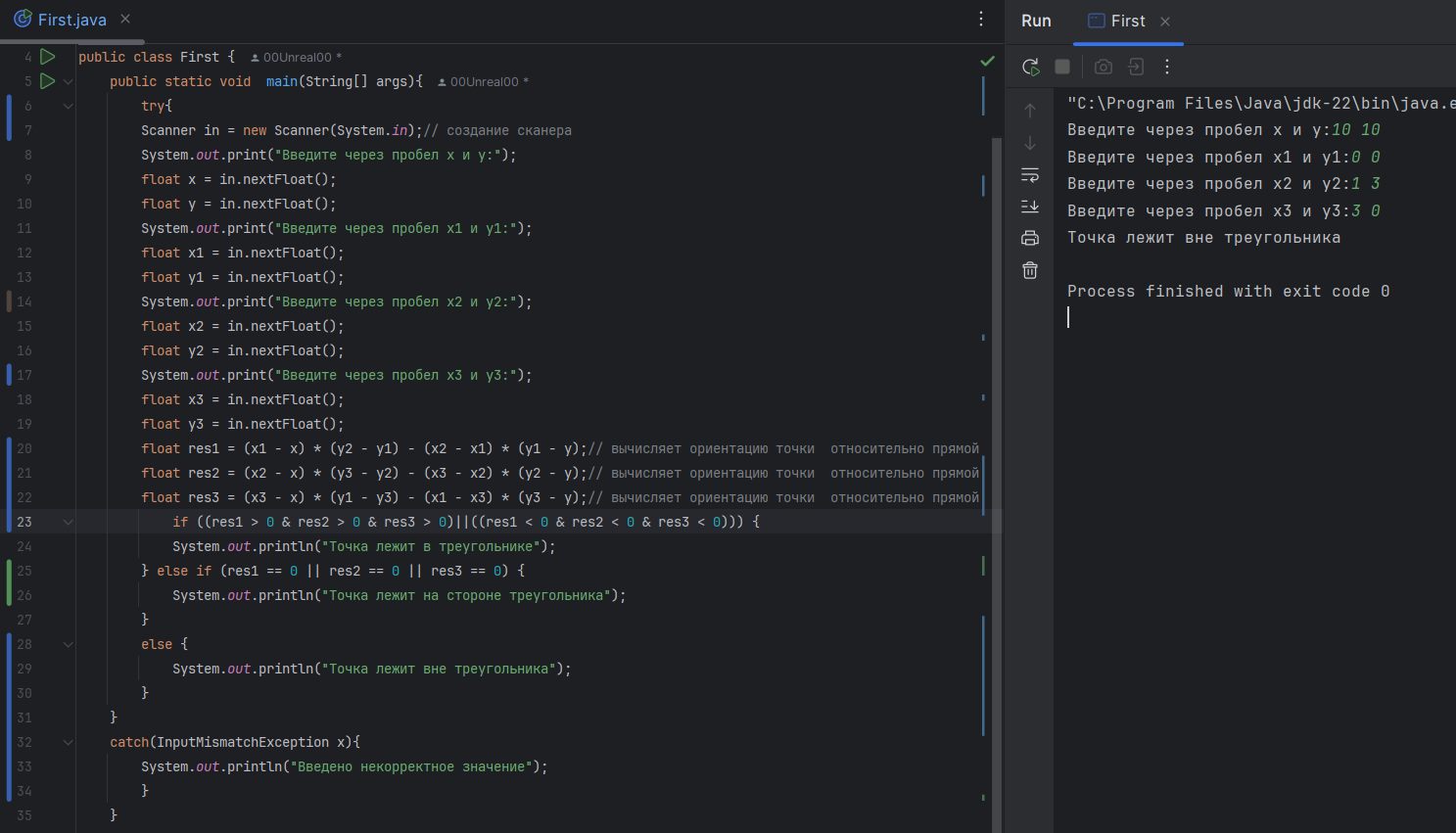


Рисунок 3 – Результат работы программы

1. В составленную программу вводятся данные, после чего в конце ее работы можно увидеть вывод одного из условий

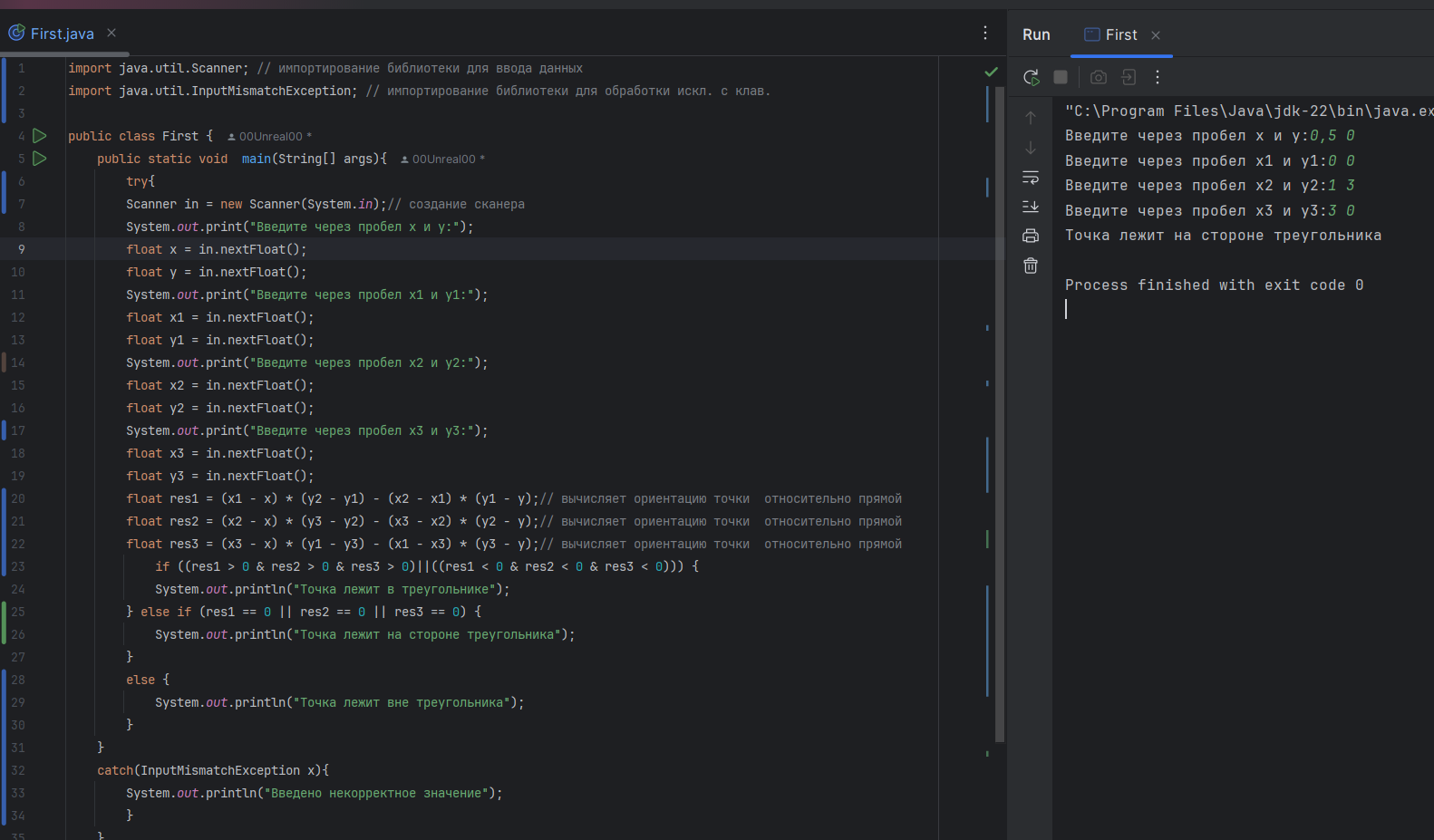


Рисунок 4 – Результат работы программы

**6** Вывод:

В ходе выполнения работы были изучены: структура программы Java, типы данных в Java, арифметические операции, условные конструкции и обработка исключений.

Было проделано: анализ возможных ситуаций, анализ возможных ошибок, написание кода и составление блок схемы

На контрольных примерах мы убедились, что программа работает корректно, и в случае ошибки ввода пользователем уведомляет его об этом.

Был оформлен комплект документации на программный код.