Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическая 7\_3**»

Выполнил: Карамов Д.Э.

Группа: ПР-24

Преподаватель: Мирошниченко Г.В

2025

**1 Задание**

**Задание:**

****

**Входные данные**

a – double

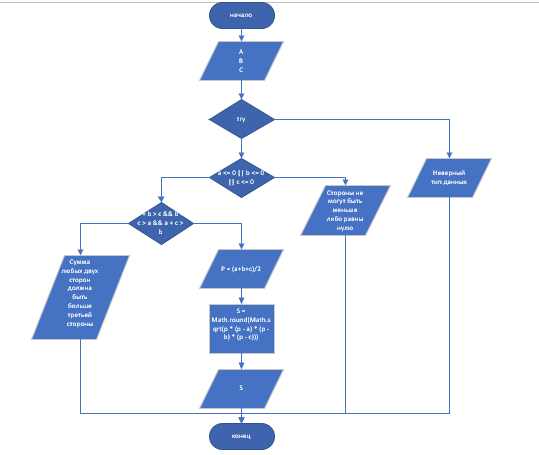
b – double

c – double

**Выходные данные**

s – double

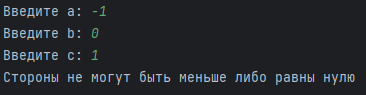
**Блок-схема**

****

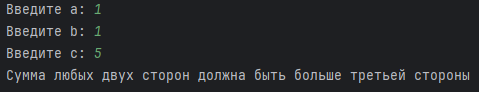
**Листинг программы**

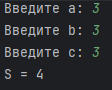
fun main()  
{  
 try  
 {  
 *print*("Введите a: ")  
 val a = *readln*().*toDouble*()  
 *print*("Введите b: ")  
 val b = *readln*().*toDouble*()  
 *print*("Введите c: ")  
 val c = *readln*().*toDouble*()  
 when  
 {  
 a <= 0 || b <= 0 || c <= 0 -> *println*("Стороны не могут быть меньше либо равны нулю")  
 a + b > c && b + c > a && a + c > b ->  
 {  
 val p = (a + b + c)/2  
 val s = Math.round(Math.sqrt(p \* (p - a) \* (p - b) \* (p - c)))  
 *println*("S = $s")  
 }  
 else -> *println*("Сумма любых двух сторон должна быть больше третьей стороны")  
 }  
 }  
 catch(e:Exception)  
 {  
 *println*("Неверный тип данных")  
 }  
}

**Тестовые ситуации**







**Результат:  
**

**2 Задание**

**Задание:**

****

**Входные данные**

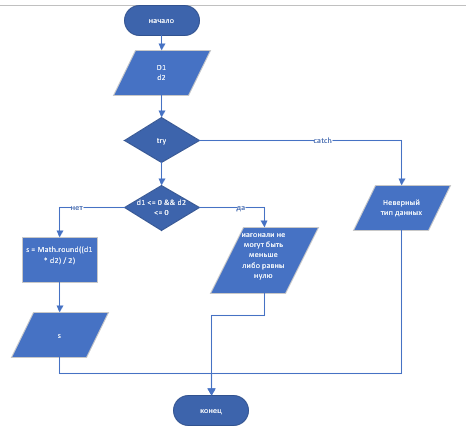
D1 – double

D2 - double

**Выходные данные**

**S - double**

**Блок-схема**

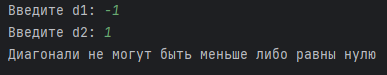


**Листинг программы**

fun main()  
{  
 try  
 {  
 *print*("Введите d1: ")  
 val d1 = *readln*().*toDouble*()  
 *print*("Введите d2: ")  
 val d2 = *readln*().*toDouble*()  
 when  
 {  
 d1 <= 0 || d2 <= 0 -> *println*("Диагонали не могут быть меньше либо равны нулю")  
 else ->  
 {  
 val s = Math.round((d1 \* d2) / 2)  
 *println*("S = $s")  
 }  
 }  
 }  
 catch(e:Exception)  
 {  
 *println*("Неверный тип данных")  
 }  
}

**Тестовые ситуации**



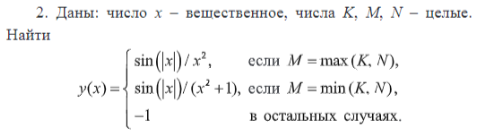


**Результат:**

****

**3 Задание**

**Задание:**

****

**Входные данные**

**X – double**

**K – Int**

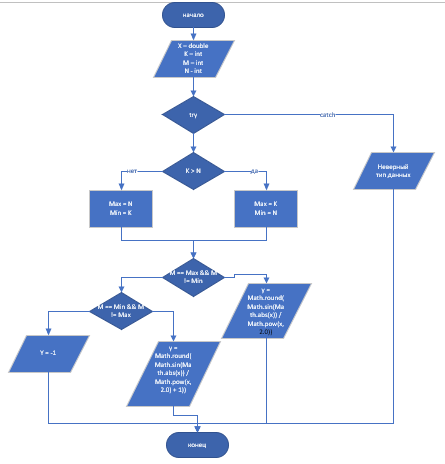
**M – Int**

**N - Int**

**Выходные данные**

**Y - double**

**Блок-схема**



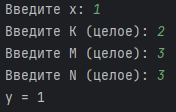
**Листинг программы**

fun main()  
{  
 try  
 {  
 *print*("Введите x: ")  
 val x = *readln*().*toDouble*()  
 *print*("Введите K (целое): ")  
 val K = *readln*().*toInt*()  
 *print*("Введите M (целое): ")  
 val M = *readln*().*toInt*()  
 *print*("Введите N (целое): ")  
 val N = *readln*().*toInt*()  
 val Max:Int  
 val Min:Int  
 val y:Long  
 when  
 {  
 K > N ->  
 {  
 Max = K; Min = N  
 }  
 else ->  
 {  
 Max = N; Min = K  
 }  
 }  
 when  
 {  
 M == Max && M != Min ->  
 {  
 y = Math.round(Math.sin(Math.abs(x)) / Math.pow(x, 2.0))  
 }  
 M == Min && M != Max ->  
 {  
 y = Math.round(Math.sin(Math.abs(x)) / (Math.pow(x, 2.0) + 1))  
 }  
 else ->  
 {  
 y = -1  
 }  
 }  
 *println*("y = $y")  
 }  
 catch(e:Exception)  
 {  
 *println*("Неверный тип данных")  
 }  
}

**Тестовые ситуации**

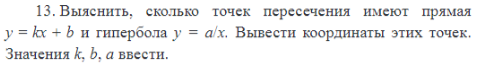


**Результат:**

****

**4 Задание**

**Задание:**

****

**Входные данные**

**K – double**

**B – double**

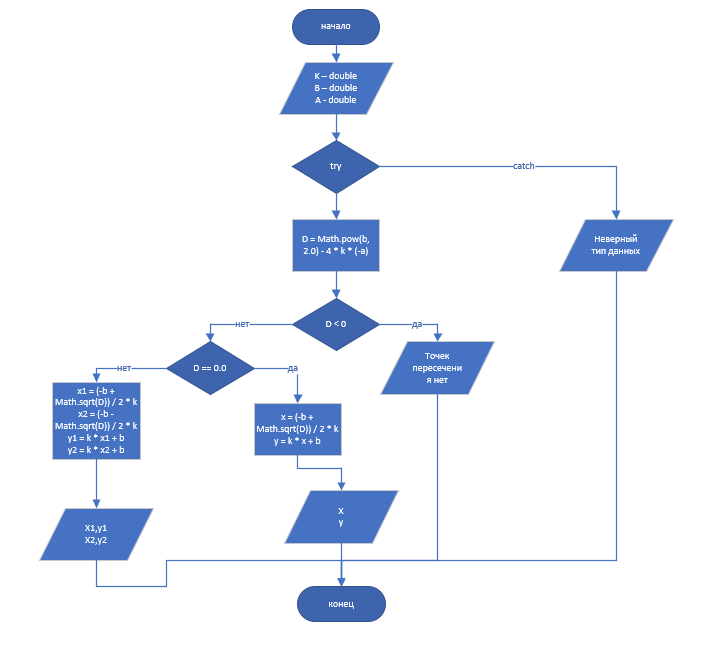
**A - double**

**Выходные данные**

X1,Y1 - double

X2,Y2 – double – Координаты точек

**Блок-схема**

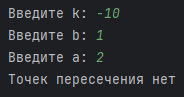
****

**Листинг программы**

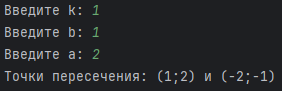
fun main()  
{  
 try  
 {  
 *print*("Введите k: ")  
 val k = *readln*().*toDouble*()  
 *print*("Введите b: ")  
 val b = *readln*().*toDouble*()  
 *print*("Введите a: ")  
 val a = *readln*().*toDouble*()  
 val D = Math.pow(b, 2.0) - 4 \* k \* (-a)  
 when  
 {  
 D < 0 -> *println*("Точек пересечения нет")  
 D == 0.0 ->  
 {  
 val x = (-b + Math.sqrt(D)) / 2 \* k  
 val y = k \* x + b  
 *println*("Точка пересечения : (${Math.round(x)};${Math.round(y)})")  
 }  
 else ->  
 {  
 val x1 = (-b + Math.sqrt(D)) / 2 \* k  
 val x2 = (-b - Math.sqrt(D)) / 2 \* k  
 val y1 = k \* x1 + b  
 val y2 = k \* x2 + b  
 *println*("Точки пересечения: (${Math.round(x1)};${Math.round(y1)}) и (${Math.round(x2)};${Math.round(y2)})")  
 }  
 }  
 }  
 catch(e:Exception)  
 {  
 *println*("Неверный тип данных")  
 }  
}

**Тестовые ситуации**



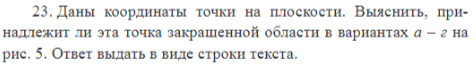


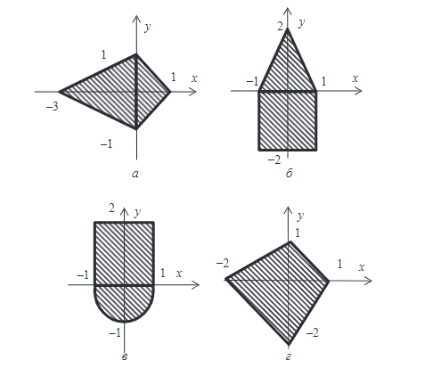
**Результат:**

****

**5 Задание**

**Задание:**

****

****

**Входные данные**

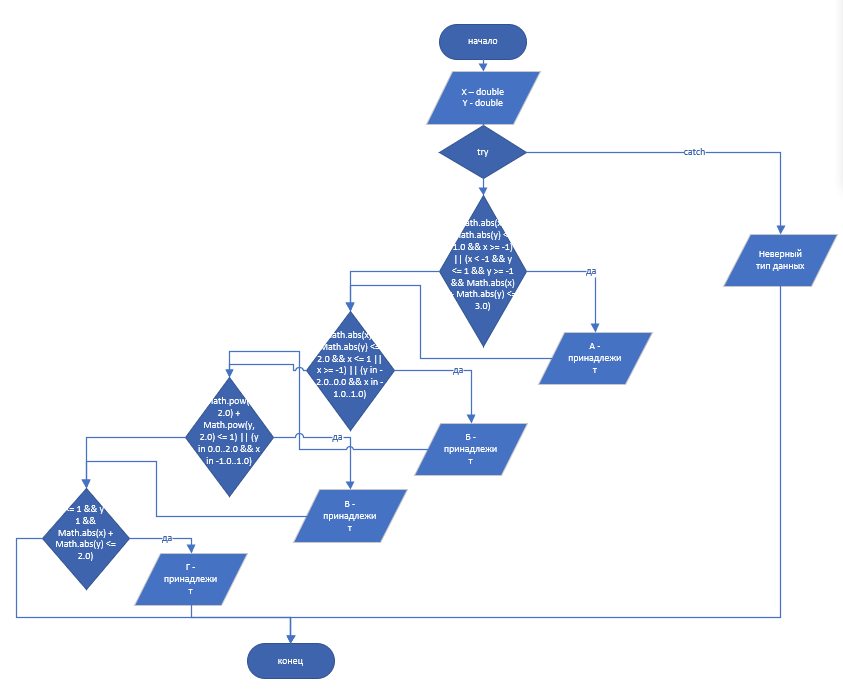
**X – double**

**Y - double**

**Выходные данные**

**“Принадлежит || не принадлежит”**

**Блок-схема**

****

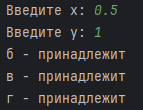
**Листинг программы**

fun main()  
{  
 try  
 {  
 *print*("Введите x: ")  
 val x = *readln*().*toDouble*()  
 *print*("Введите y: ")  
 val y = *readln*().*toDouble*()  
 when  
 {  
 (Math.abs(x) + Math.abs(y) <= 1.0 && x >= -1) || (x < -1 && y <= 1 && y >= -1 && Math.abs(x) + Math.abs(y) <= 3.0) ->  
 {  
 *println*("a - принадлежит")  
 }  
 }  
 when  
 {  
 (Math.abs(x) + Math.abs(y) <= 2.0 && x <= 1 || x >= -1) || (y in -2.0..0.0 && x in -1.0..1.0) ->  
 {  
 *println*("б - принадлежит")  
 }  
 }  
 when  
 {  
 (Math.pow(x, 2.0) + Math.pow(y, 2.0) <= 1) || (y in 0.0..2.0 && x in -1.0..1.0) ->  
 {  
 *println*("в - принадлежит")  
 }  
 }  
 when  
 {  
 (x <= 1 && y <= 1 && Math.abs(x) + Math.abs(y) <= 2.0) ->  
 {  
 *println*("г - принадлежит")  
 }  
 }  
 }  
 catch(e:Exception)  
 {  
 *println*("Неверный тип данных")  
 }  
}

**Тестовые ситуации**



**Результат:**

****