Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчет по программе «Практическое занятие 7.1»

Выполнил: Меньшиков Артур Андреевич

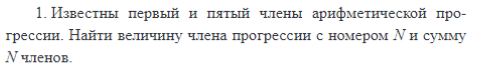
Группа: ПР-22

Преподаватель: Мирошниченко Г.В

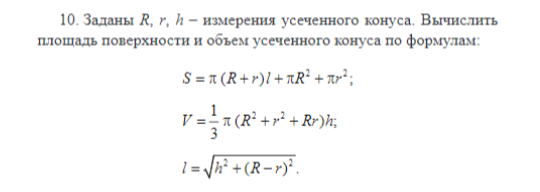
2023

**Задание:**

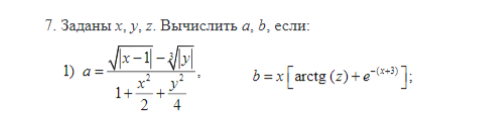
**задание 1:**

****

**задание 2:**

****

**задание 3:**

****

**Входные и выходные данные**

**задание 1:**

a - int/целочисленный тип

b - int/целочисленный тип

n - int/целочисленный тип

**задание 2:**

R - double/ число с плавающей запятой

r - double/ число с плавающей запятой

h - double/ число с плавающей запятой

**задание 3:**

x - double/ число с плавающей запятой

y - double/ число с плавающей запятой

z - double/ число с плавающей запятой

**Блок-схема**

**…**

**Листинг программы (если есть)**

**задание 1:**

**fun main(){**

**try{**

***println*("введите первый член прогрессии")**

**var a = *readln*()!!.*toInt*() //первый член прогрессии**

***println*("введите второй член прогрессии")**

**var b = *readln*()!!.*toInt*() //второй член прогрессии**

***println*("введите номер искомого члена прогрессии")**

**var n = *readln*()!!.*toInt*() // номер искомого члена прогрессии**

**var d = b - a**

**var an = a + (n-1)\*d**

**var sn = (n/2)\*(2\*a+(n-1)\*d)**

***println* ("Член прогрессии с номером ${n}: ${an}")**

***println* ("Сумма ${n} членов прогрессии: ${sn}")**

**} catch (e:Exception){*println*("неверный формат")}**

**}**

**задание 2:**

**import kotlin.math.\***

**fun main(){**

**try {**

***println*("введите R")**

**var R = *readln*()!!.*toDouble*()**

***println*("введите r")**

**var r = *readln*()!!.*toDouble*()**

***println*("введите h")**

**var h = *readln*()!!.*toDouble*()**

**if(R>r){**

**var v = (1/3) \* *PI* \* (R.*pow*(2) + r.*pow*(2) + R \* r) \* h**

**var l = *sqrt*(h.*pow*(2) + (R - r).*pow*(2))**

**var s = *PI* \* (R + r) \* l + *PI* \* R.*pow*(2) + *PI* \* r.*pow*(2)**

***println*("S = ${String.*format*("%.2f",s)}; V = ${String.*format*("%.2f",v)}; l = ${String.*format*("%.2f",l)}")**

**}**

**else *println*("радиус нижнего основания (R) не может быть меньше радиуса верхнего основания (r)")**

**} catch (e:Exception){*println*("неверный формат")}**

**}**

**задание 3:**

**import kotlin.math.\***

**fun main(){**

**try {**

***println*("введите x")**

**var x = *readln*()!!.*toDouble*()**

***println*("введите y")**

**var y = *readln*()!!.*toDouble*()**

**var y1 = y.toDouble().*pow*(1/3.toDouble())**

***println*("введите z")**

**var z = *readln*()!!.*toDouble*()**

**var a = (*sqrt*((*abs*(x-1))- *abs*(y1))) / (1+(x.*pow*(2)/2)+(y.*pow*(2)/4))**

**var b = x\*(*atan*(z)+ *E*.*pow*(-(x+3)))**

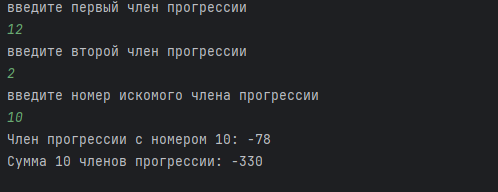
***println*("a = ${a}; b = ${b}")**

**}catch (e:Exception){*println*("неверный формат")}**

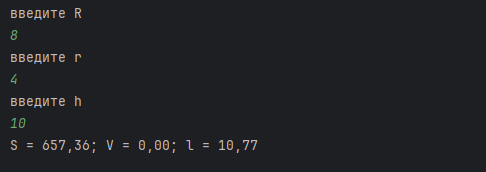
**}**

**Тестовые ситуации**

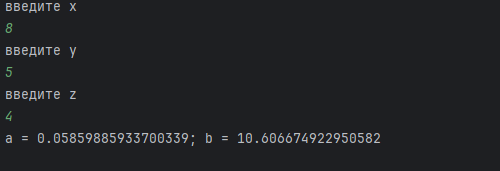
**задание 1:**

****

**Задание 2:**

****

**Задание 3:**

****