Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

# Отчёт по программе «Практическое занятие 10»

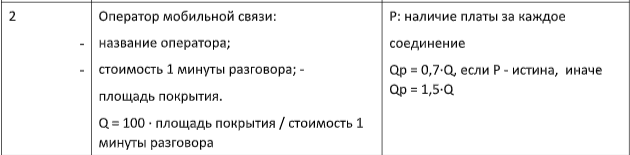
Выполнил: Юнусов Руслан Александрович

Группа: ПР-23

Преподаватель: Мирошниченко Г.В

2024

**Задание:**

****

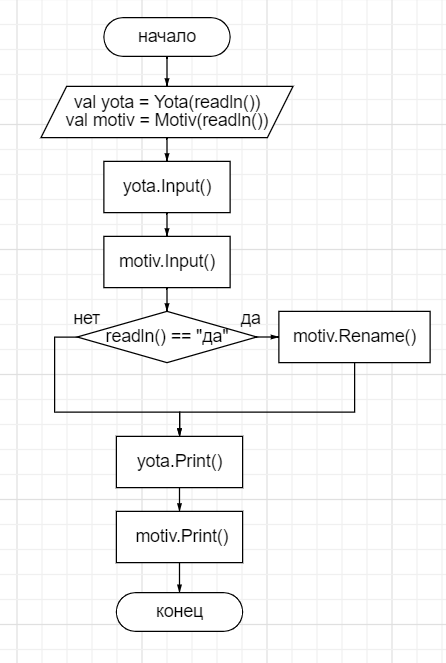
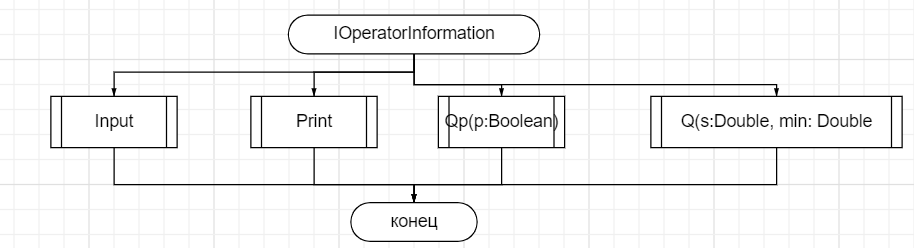
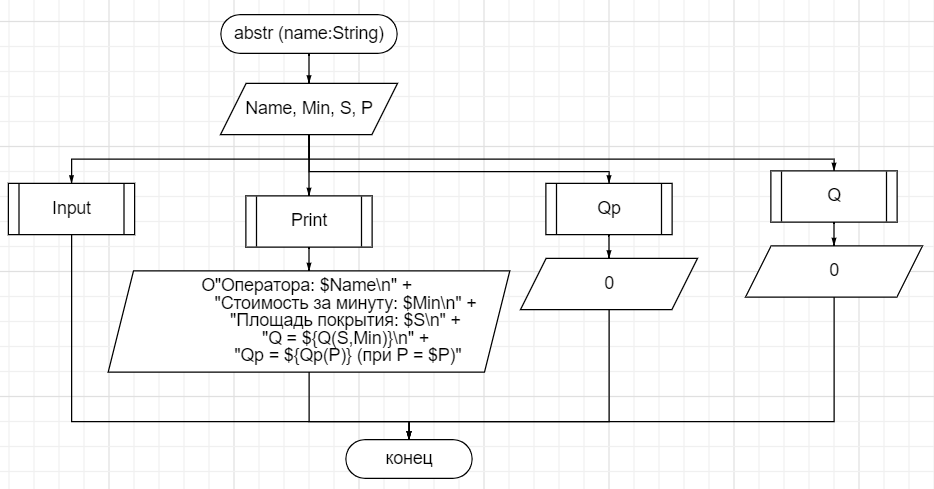
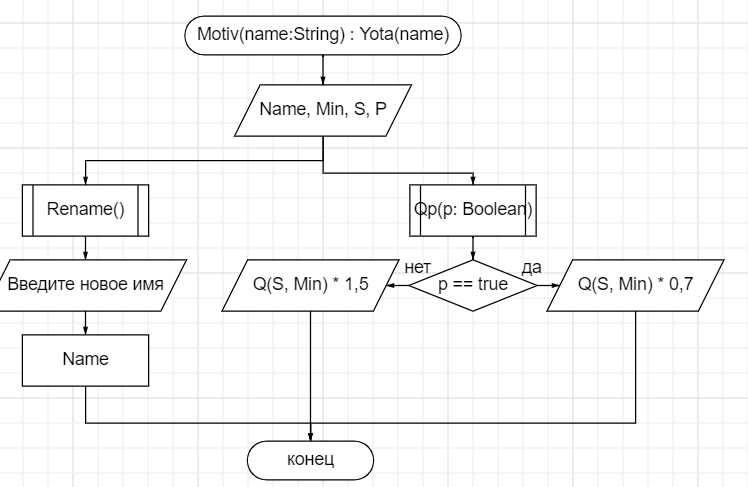
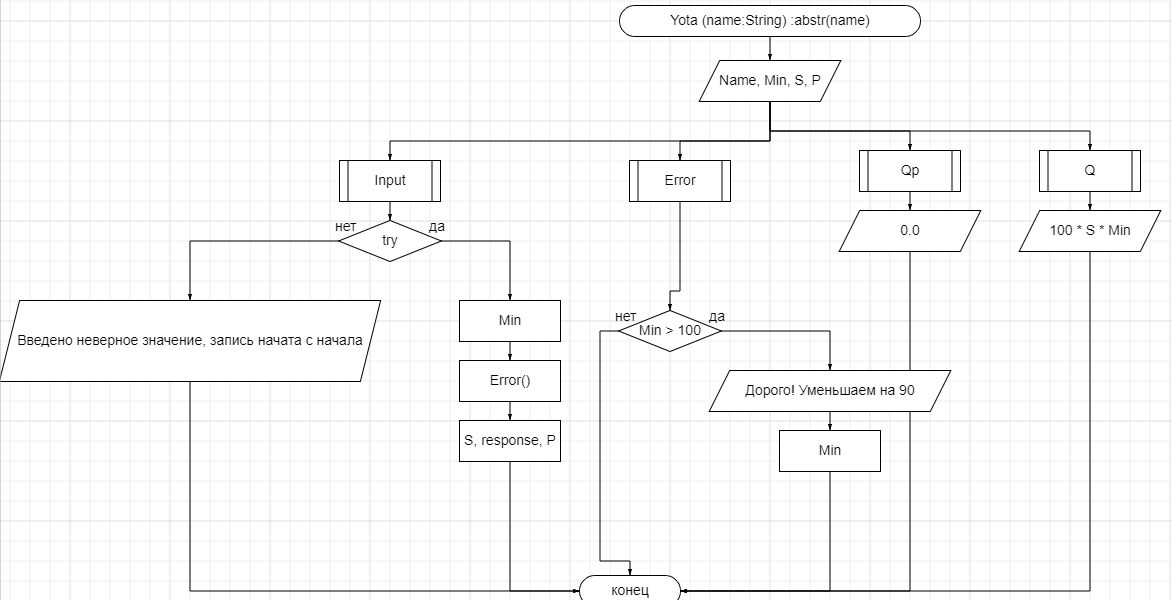
**Входные и выходные данные**

Name – string, строка, название оператора

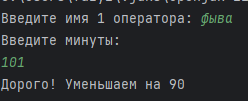
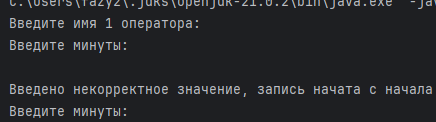
Min – double, число с плавающей запятой, стоимость за минуту

S - double, число с плавающей запятой, площадь покрытия

P – boolean, булево значение, оплата за соединение

**Блок-схема  
**

**Тестовые ситуации**



**Вывод**

Я улучшие свои навыки работы с языком программирования Kotlin

Листинг программы:

fun main() {  
 *print*("Введите имя 1 оператора: ")  
 val yota = Yota(*readln*())  
 yota.Input()  
  
 *print*("Введите имя 2 оператора: ")  
 val motiv = Motiv(*readln*())  
 motiv.Input()  
  
 *print*("Хотите переименовать оператор (да/нет): ")  
 if (*readln*() == "да"){  
 motiv.Rename()  
 }  
  
 *println*("----------------------------------------------------")  
 *println*("Оператор 1:")  
 yota.Print()  
  
 *println*("----------------------------------------------------")  
 *println*("Оператор 2:")  
 motiv.Print()  
}

interface IOperatorInformation {  
 fun Input()  
 fun Print()  
 fun Qp(p:Boolean):Double  
 fun Q(s:Double,min:Double):Double  
}

abstract class abstr (name:String) : IOperatorInformation {  
 open var Name = name  
 abstract var Min:Double  
 abstract var S:Double  
 abstract var P:Boolean  
  
 override fun Input() {}  
  
 override fun Print() {  
 *println*("Оператора: $Name\n" +  
 "Стоимость за минуту: $Min\n" +  
 "Площадь покрытия: $S\n" +  
 "Q = ${Q(S,Min)}\n" +  
 "Qp = ${Qp(P)} (при P = $P)"  
 )  
 }  
 override fun Qp(p:Boolean):Double {  
 return 0.0  
 }  
 override fun Q(s:Double,min:Double):Double {  
 return 0.0  
 }  
}

open class Yota (name:String) :abstr(name) {  
 override var Name = name  
 override var Min = 0.0  
 override var S = 0.0  
 override var P = false  
  
 override fun Input() {  
 super.Input()  
 try {  
 *println*("Введите минуты:")  
 Min = *readln*().*toDouble*()  
 Error()  
  
 *println*("Введите стоимость:")  
 S = *readln*().*toDouble*()  
  
 *println*("Есть ли оплата за каждое соединение? (да/нет)")  
 val response = *readln*()  
 P = response.*equals*("да", ignoreCase = true)  
 } catch (a: Exception) {  
 *println*("Введено некорректное значение, запись начата с начала")  
 Input()  
 }  
 }  
  
 fun Error(){  
 if (Min>100.0){  
 *println*("Дорого! Уменьшаем на 90")  
 Min-=90.0  
 }  
 }  
 override fun Qp(p:Boolean):Double{  
 return 0.0  
 }  
 override fun Q(s:Double,min:Double):Double{  
 return 100\*S\*Min  
 }  
}

class Motiv(name:String) : Yota(name){  
 override var Name = name  
 override var Min = 0.0  
 override var S = 0.0  
 override var P = false  
  
 fun Rename(){  
 *println*("Введите новое имя")  
 Name = *readln*()  
 }  
 override fun Qp(p:Boolean):Double{  
 if (p){  
 return Q(S,Min)\*0.7  
 }  
 else{  
 return Q(S,Min)\*1.5  
 }  
 }  
}