МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ПЕРМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Электротехнический факультет

Кафедра ИТАС

II семестр

“Основы алгоритмизации и программирования”

**Задача коммивояжера**

Выполнил

студент группы РИС-20-2б

Уржумов В.И.

Проверила

доцент кафедры ИТАС

Полякова О.А.

Пермь, 2021

**Цель работы**

Создать алгоритм и визуализацию задачи коммивояжера.

**Постановка задачи**

Имеется N городов, которые должен обойти коммивояжер с минимальными затратами. При этом на его маршрут накладывается два ограничения:

* маршрут должен быть замкнутым, то есть коммивояжер должен вернуться в тот город, из которого он начал движение;
* в каждом из городов коммивояжер должен побывать точно один раз, то есть надо обязательно обойти все города, при этом, не побывав ни в одном городе дважды.

**Анализ задачи**

1. Реализовать алгоритм нахождения кратчайшего пути.
2. Визуализировать введенный граф с помощью OpenGL.
3. Реализовать возможность добавлять, удалять элементы графа.

**Ход работы**

1 Создание пользовательского интерфейса

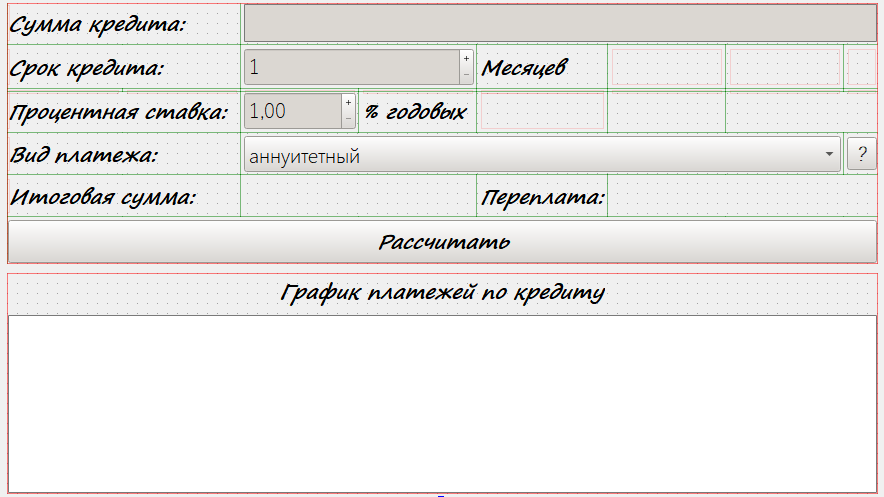
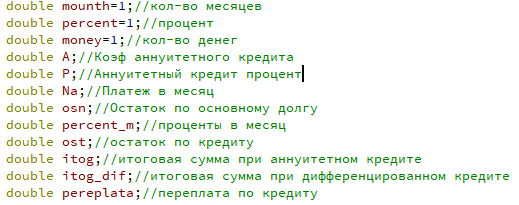


Таблица цветов, использованных в калькуляторе

|  |  |
| --- | --- |
| #DBD7D2 |  |
| #B8C2E3 |  |

2 Определение необходимых переменных



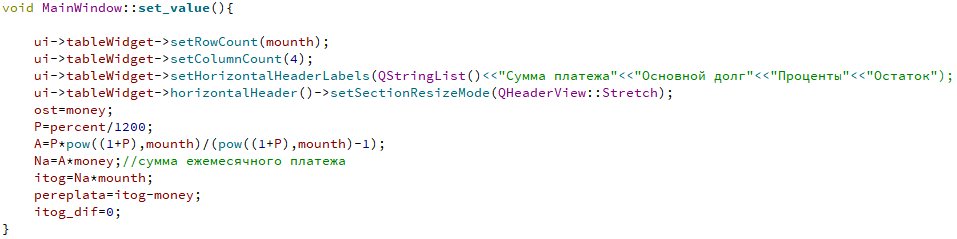
3Определение метода вычисления аннуитетного кредита



4Определение метода вычисления дифференциального кредита



5Определение метода инициализации переменных



**Листинг программы**

**Mainwindow.h**

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

QT\_BEGIN\_NAMESPACE

namespace **Ui** { class **MainWindow**; }

QT\_END\_NAMESPACE

class **MainWindow** : public QMainWindow

{

Q\_OBJECT

public:

**MainWindow**(QWidget \*parent = nullptr);

~***MainWindow***();

private:

double mounth=1;//кол-во месяцев

double percent=1;//процент

double money=1;//кол-во денег

double A;//Коэф аннуитетного кредита

double P;//Аннуитетный кредит процент

double Na;//Платеж в месяц

double osn;//Остаток по основному долгу

double percent\_m;//проценты в месяц

double ost;//остаток по кредиту

double itog;//итоговая сумма при аннуитетном кредите

double itog\_dif;//итоговая сумма при дифференцированном кредите

double pereplata;//переплата по кредиту

Ui::MainWindow \*ui;

void **anuit**();//вычисление аннуитетного

void **dif**();//вычисление дифференцированного

void **set\_value**();//установка значений

private slots:

void **on\_spinBox\_valueChanged**(int arg1);

void **on\_pushButton\_clicked**();

void **on\_lineEdit\_textChanged**(const QString &arg1);

void **on\_pushButton\_2\_clicked**();

void **on\_doubleSpinBox\_valueChanged**(double arg1);

};

#endif // MAINWINDOW\_H

**Mainwindows.cpp**

#include "mainwindow.h"

#include "ui\_mainwindow.h"

#include<QtMath>

#include<QMessageBox>

MainWindow::**MainWindow**(QWidget \*parent)

: QMainWindow(*parent*)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

this->setStyleSheet("background-color:#b8c2e3;");

ui->lineEdit->setValidator(new QIntValidator(0,1000000,this));

}

MainWindow::~***MainWindow***()

{

delete ui;

}

void MainWindow::**set\_value**(){

ui->tableWidget->setRowCount(mounth);

ui->tableWidget->setColumnCount(4);

ui->tableWidget->setHorizontalHeaderLabels(QStringList()<<"Сумма платежа"<<"Основной долг"<<"Проценты"<<"Остаток");

ui->tableWidget->horizontalHeader()->setSectionResizeMode(QHeaderView::Stretch);

ost=money;

P=percent/1200;

A=P\*pow((1+P),mounth)/(pow((1+P),mounth)-1);

Na=A\*money;//сумма ежемесячного платежа

itog=Na\*mounth;

pereplata=itog-money;

itog\_dif=0;

}

//аннуитетный

void MainWindow::**anuit**(){

set\_value();//установка значений

ui->label\_22->setText(QString::number(itog,'g',7));//вывод итоговой суммы кредита

ui->label\_23->setText(QString::number(pereplata,'g',7));//вывод переплаты

for(int i=0;i<ui->tableWidget->rowCount();i++){

percent\_m=ost\*P;//вычисление процента

osn=Na-percent\_m;//вычисление основоного долга

ost-=osn;//вычисление остатка

if(ost<=1){

ost=0;

}

QTableWidgetItem\* item1 = new QTableWidgetItem;//в месяц

QTableWidgetItem\* item2 = new QTableWidgetItem;//основной долг

QTableWidgetItem\* item3 = new QTableWidgetItem;//процент в месяц

QTableWidgetItem\* item4 = new QTableWidgetItem;//остаток

QString sum=QString::number(Na,'g',5);

QString osn1=QString::number(osn,'g',5);

QString pm=QString::number(percent\_m,'g',4);

QString ost1=QString::number(ost,'g',5);

item1->setText(sum);

item2->setText(osn1);

item3->setText(pm);

item4->setText(ost1);

//добавление в таблицу

ui->tableWidget->setItem(i,0,*item1*);

ui->tableWidget->setItem(i,1,*item2*);

ui->tableWidget->setItem(i,2,*item3*);

ui->tableWidget->setItem(i,3,*item4*);

}

}

//дифференцированный

void MainWindow::**dif**(){

set\_value();//установка значений

ui->tableWidget->setRowCount(mounth);

for(int i=0;i<mounth;i++){

double telo=money/mounth;//основной долг

double ostatok=money-(telo\*(i));//остаток задолжности

double proc=ostatok\*(percent/1200);//процент в месяц

double in\_mounth=telo+proc;//процент в месяц

QTableWidgetItem\* item1 = new QTableWidgetItem;//в месяц

QTableWidgetItem\* item2 = new QTableWidgetItem;//основной долг

QTableWidgetItem\* item3 = new QTableWidgetItem;//процент в месяц

QTableWidgetItem\* item4 = new QTableWidgetItem;//остаток

QString in=QString::number(in\_mounth,'g',5);

QString osn1=QString::number(telo,'g',5);

QString pm=QString::number(proc,'g',4);

QString ost1=QString::number(ostatok,'g',5);

item1->setText(in);

item2->setText(osn1);

item3->setText(pm);

item4->setText(ost1);

//добавление в таблицу

ui->tableWidget->setItem(i,0,*item1*);

ui->tableWidget->setItem(i,1,*item2*);

ui->tableWidget->setItem(i,2,*item3*);

ui->tableWidget->setItem(i,3,*item4*);

itog\_dif+=in\_mounth;

}

pereplata=itog\_dif-money;

ui->label\_22->setText(QString::number(itog\_dif,'g',7));//вывод итоговой суммы кредита

ui->label\_23->setText(QString::number(pereplata,'g',7));//вывод переплаты

}

void MainWindow::**on\_spinBox\_valueChanged**(int arg1)

{

mounth=arg1;//при изменении==присвоение месяца

}

//выбор вида кредита

void MainWindow::**on\_pushButton\_clicked**(){

int i=ui->comboBox->currentIndex();

switch(i){

case 0: anuit();

break;

case 1: dif();

break;

}

}

void MainWindow::**on\_lineEdit\_textChanged**(const QString &arg1)

{

money=arg1.toDouble();//при изменении==присвоение кол-во денег

}

//Вывод справки по видам кредита

void MainWindow::**on\_pushButton\_2\_clicked**()

{

int i=ui->comboBox->currentIndex();

if(i==0){

QMessageBox msg;

msg.setWindowTitle("Справка");

msg.setText("Аннуитетный платеж- метод погашения, при котором размер ссуды делится на равные части,которые клиент стабильно выплачивает");

msg.*exec*();

}

else{

QMessageBox msg;

msg.setWindowTitle("Справка");

msg.setText("Дифференциальный платеж- метод погашения, при котором заемщик внисит разные суммы, размер которых с каждым разом уменьшается");

msg.*exec*();

}

}

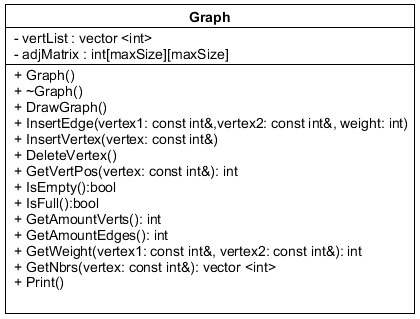
void MainWindow::**on\_doubleSpinBox\_valueChanged**(double arg1)

{

percent=arg1;//при изменении==присвоение кол-во процентов

}

**Диаграмма классов**



**Скриншоты**

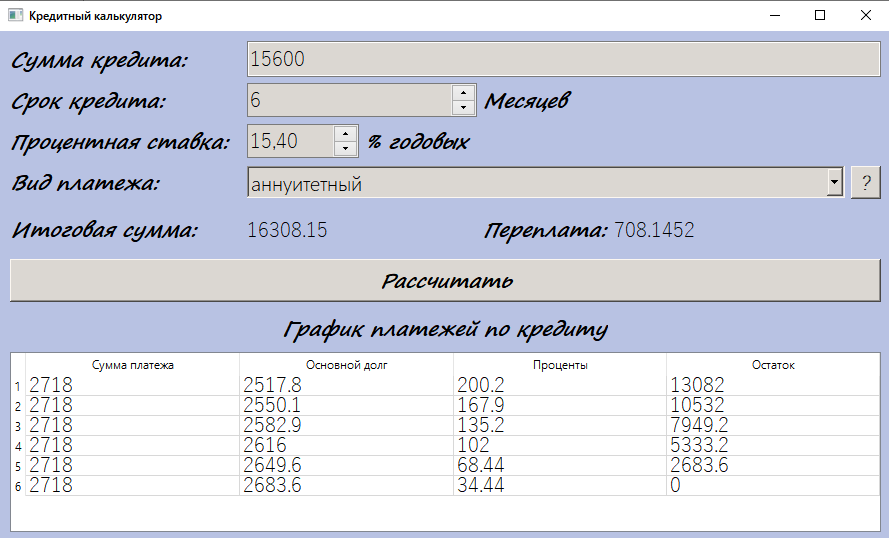


Рисунок 1-Работа кредитного калькулятора



Рисунок 2-Данные для проверки правильности