Міністерство освіти і науки України

КПІ ім. Ігоря Сікорського

Кафедра ІПІ

**ЗВІТ**

з виконання лабораторної роботи № 7

з кредитного модуля

“Основи програмування-2. Методології програмування”

**Варіант № 5**

Виконав:

Студент 1-го курсу

Групи ІП-22 ФІОТ

Гринько Олександр Максимович

Київ 2023

**Постановка задачі**

Изображение выглядит как текст, Шрифт, белый, чек

Автоматически созданное описание

**Вимоги до програми**

Изображение выглядит как текст, Шрифт

Автоматически созданное описание

**Код програми**

**main.cpp**

#include "mainwindow.h"

#include "./ui\_mainwindow.h"

#include "class.h"

#include "class.cpp"

#include <QDoubleValidator>

MainWindow::MainWindow(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

setFixedSize(700, 400);

QDoubleValidator \*validator = new QDoubleValidator(this);

validator->setLocale(QLocale(QLocale::English));

ui->x1Value->setValidator(validator);

ui->x2Value->setValidator(validator);

ui->n1Value->setValidator(validator);

ui->n2Value->setValidator(validator);

ui->m1Value->setValidator(validator);

ui->m2Value->setValidator(validator);

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::on\_countBut\_clicked()

{

double result, x1, x2, n1, n2, m1, m2;

try {

ui->labelResultValue->clear();

if (ui->x1Value->text().size() == 0 || ui->x2Value->text().size() == 0 || ui->m1Value->text().size() == 0 || ui->m2Value->text().size() == 0 || ui->n1Value->text().size() == 0 || ui->n2Value->text().size() == 0)

throw QString("Заповніть усі поля, будь ласка!");

expression A;

expression B;

x1 = ui->x1Value->text().toDouble();

A.setX(x1);

n1 = ui->n1Value->text().toDouble();

m1 = ui->m1Value->text().toDouble();

x2 = ui->x2Value->text().toDouble();

B.setX(x2);

n2 = ui->n2Value->text().toDouble();

m2 = ui->m2Value->text().toDouble();

if (n1 == m1 || n2 == m2) {

throw QString("n не може дорівнювати m!");

}

A.setN(n1);

A.setM(m1);

B.setN(n2);

B.setM(m2);

result = A.express() + B.express();

ui->labelResultValue->setText(QString::number(result));

} catch (QString error\_message) {

QMessageBox::critical(this, "Помилка вводу", error\_message);

return;

}

}

void MainWindow::on\_clearBut\_clicked()

{

ui->x1Value->clear();

ui->x2Value->clear();

ui->n1Value->clear();

ui->n2Value->clear();

ui->m1Value->clear();

ui->m2Value->clear();

ui->labelResultValue->clear();

}

**class.h**

#ifndef CLASS\_H

#define CLASS\_H

#include <iostream>

#include <cmath>

using namespace std;

class expression

{

private:

double n;

double m;

double x;

public:

double getN();

double getM();

double getX();

void setN(double n1);

void setM(double m1);

void setX(double x1);

double express();

};

#endif // CLASS\_H

**class.cpp**

#include "class.h"

void expression::setX(double x1) {

x = x1;

}

void expression::setN(double n1) {

n = n1;

}

void expression::setM(double m1) {

m = m1;

}

double expression::getX() {

return x;

}

double expression::getN() {

return n;

}

double expression::getM() {

return m;

}

double expression::express() {

double a = pow(atan(x + 1), 2) - (n / (n - m));

return a;

}

**mainwindow.h**

#ifndef MAINWINDOW\_H

#define MAINWINDOW\_H

#include <QMainWindow>

#include <QMessageBox>

QT\_BEGIN\_NAMESPACE

namespace Ui { class MainWindow; }

QT\_END\_NAMESPACE

class MainWindow : public QMainWindow

{

Q\_OBJECT

public:

MainWindow(QWidget \*parent = nullptr);

~MainWindow();

private slots:

void on\_countBut\_clicked();

void on\_clearBut\_clicked();

private:

Ui::MainWindow \*ui;

};

#endif // MAINWINDOW\_H

**mainwindow.cpp**

#include "mainwindow.h"

#include "./ui\_mainwindow.h"

#include "class.h"

#include "class.cpp"

#include <QDoubleValidator>

MainWindow::MainWindow(QWidget \*parent)

: QMainWindow(parent)

, ui(new Ui::MainWindow)

{

ui->setupUi(this);

setFixedSize(700, 400);

QDoubleValidator \*validator = new QDoubleValidator(this);

validator->setLocale(QLocale(QLocale::English));

ui->x1Value->setValidator(validator);

ui->x2Value->setValidator(validator);

ui->n1Value->setValidator(validator);

ui->n2Value->setValidator(validator);

ui->m1Value->setValidator(validator);

ui->m2Value->setValidator(validator);

}

MainWindow::~MainWindow()

{

delete ui;

}

void MainWindow::on\_countBut\_clicked()

{

double result, x1, x2, n1, n2, m1, m2;

try {

ui->labelResultValue->clear();

if (ui->x1Value->text().size() == 0 || ui->x2Value->text().size() == 0 || ui->m1Value->text().size() == 0 || ui->m2Value->text().size() == 0 || ui->n1Value->text().size() == 0 || ui->n2Value->text().size() == 0)

throw QString("Заповніть усі поля, будь ласка!");

expression A;

expression B;

x1 = ui->x1Value->text().toDouble();

A.setX(x1);

n1 = ui->n1Value->text().toDouble();

m1 = ui->m1Value->text().toDouble();

x2 = ui->x2Value->text().toDouble();

B.setX(x2);

n2 = ui->n2Value->text().toDouble();

m2 = ui->m2Value->text().toDouble();

if (n1 == m1 || n2 == m2) {

throw QString("n не може дорівнювати m!");

}

A.setN(n1);

A.setM(m1);

B.setN(n2);

B.setM(m2);

result = A.express() + B.express();

ui->labelResultValue->setText(QString::number(result));

} catch (QString error\_message) {

QMessageBox::critical(this, "Помилка вводу", error\_message);

return;

}

}

void MainWindow::on\_clearBut\_clicked()

{

ui->x1Value->clear();

ui->x2Value->clear();

ui->n1Value->clear();

ui->n2Value->clear();

ui->m1Value->clear();

ui->m2Value->clear();

ui->labelResultValue->clear();

}

**Результат роботи**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

**Гіт хаб: https://github.com/esk4nz/OP\_labs**