

Міністерство освіти і науки України

КПІ ім. Ігоря Сікорського

Кафедра ІІІ

ЗВІТ

з виконання лабораторної роботи № 7-8

з кредитного модуля

“Основи програмування-2. Методології програмування”

Варіант № 21

Виконав:

студент 1-го курсу

гр. ІІ-22 ФІОТ

Патріюк Юрій Олексійович

Київ 2023

## ПОСТАНОВКА ЗАДАЧІ

21. Розробити клас "Файл чисел" для представлення бінарного файлу, що містить цілі додатні числа. Створити бінарний файл (об'єкт даного класу). На його основі створити новий файл, в якому зберегти числа, які належать заданому діапазону, збільшивши їх у вказану кількість разів. При переповненні зверху (overflow) або при відсутності таких чисел у файлі, згенерувати відповідні виняткові ситуації і організувати їх обробку.

## ТЕКСТ ПРОГРАМИ

### Вміст файлу main.cpp

```
#include "mainwindow.h"
#include <QApplication>

int main(int argc, char *argv[])
{
    QApplication a(argc, argv);
    MainWindow w;
    w.show();
    return a.exec();
}
```

### Вміст файлу 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);
    void updateTextBrowser(std::string);
    ~MainWindow();

private slots:
```

```

        void on_saveButton_clicked();
        void on_findButton_clicked();

private:
    Ui::MainWindow *ui;
};
#endif // MAINWINDOW_H };

```

## Вміст файлу mainwindow.cpp

```

#include "mainwindow.h"
#include "numbersfile.cpp"
#include "../ui_mainwindow.h"

MainWindow::MainWindow(QWidget *parent)
    : QMainWindow(parent)
    , ui(new Ui::MainWindow)
{
    ui->setupUi(this);
}

MainWindow::~MainWindow()
{
    delete ui;
}

void MainWindow::updateTextBrowser(string name)
{
    ifstream i_file(name, ios::binary);
    size_t num;
    string s = "";
    i_file.read((char*)&num, sizeof(size_t));
    do{
        s += to_string(num) + " ";
        i_file.read((char*)&num, sizeof(size_t));
    }while(!i_file.eof());
    ui->fileTextBrowser->setText(QString::fromStdString(s));
    i_file.close();
}

```

```

void MainWindow::on_saveButton_clicked()
{
    string name = ui->fileNameLineEdit->text().toString();
    size_t size = (size_t)ui->fileSizeLineEdit->text().toInt();
    NumbersFile nfile(name, size);

    bool res = nfile.insertRange((size_t)ui->startLineEdit-
>text().toInt(), (size_t)ui->endLineEdit->text().toInt(), (size_t)ui->powerLineEdit-
>text().toInt());

    if(res)
        updateTextBrowser(name);

}

void MainWindow::on_findButton_clicked()
{
    if(ui->findLineEdit->text().isEmpty()){
        MessageBoxA(NULL, (char*)"Enter element to find!", (char*)"Error",
MB_ICONINFORMATION);
        return;
    }

    try{

        ifstream i_file(ui->fileNameLineEdit->text().toString(), ios::binary);
        if(i_file.is_open())
        {
            size_t num;
            int position = 1;
            i_file.read((char*)&num, sizeof(size_t));
            do{
                if(ui->findLineEdit->text().toInt() == num)
                {
                    string message = "Element was found at position " +
to_string(position);
                    MessageBoxA(NULL, message.c_str(), (char*)"Success",
MB_ICONINFORMATION);
                    i_file.close();
                    return;
                }
            }
        }
    }
}

```

```

        position++;

        i_file.read((char*)&num, sizeof(size_t));
    }while(!i_file.eof());

    i_file.close();

    throw NotFoundException();

}

}

catch(NotFoundException e){

    MessageBoxA(NULL, e.what(), (char*)"Error", MB_ICONWARNING);

}

}

```

## Вміст файлу numbersfile.h

```

#include <iostream>
#include <fstream>
#include <windows.h>

using namespace std;

class OverflowException: public exception{
    char* message = (char*)"Overflow!";
public:
    char* what()
    {
        return message;
    }
};

class NotFoundException: public exception{
    char* message = (char*)"Element wasn't found!";
public:
    char* what()
    {
        return message;
    }
};

```

```

class NumbersFile
{
    string file;
    size_t size;
public:
    NumbersFile(string f, size_t size): file(f), size(size){};
    bool insertRange(size_t begin, size_t end, size_t power);
};

```

### Вміст файлу numbersfile.cpp

```

#include "numbersfile.h"

bool NumbersFile::insertRange(size_t begin, size_t end, size_t power)
{
    ofstream o_file(this->file, ios_base::binary);
    try{
        int counter = 0;
        for (size_t var = begin; var != end+1; ++var) {
            counter++;
            if(counter == this->size+1)
                throw OverflowException();
            size_t temp = var * power;
            o_file.write((char*)&temp, sizeof(size_t));
        }
    }
    catch(OverflowException e){
        o_file.close();
        MessageBoxA(NULL, e.what(), (char*)"Error", MB_ICONWARNING);
        return false;
    }
    o_file.close();
    return true;
}

```

## РЕЗУЛЬТАТ ТЕСТУВАННЯ

Лабораторна 7

Назва файлу:

Розмір файлу:

Початок діапазону:

Кінець діапазону:

Множник:

Success

Element was found at position 4

Лабораторна 7

Назва файлу:

Розмір файлу:

Початок діапазону:

Кінець діапазону:

Множник:

Element wasn't found!

Лабораторна 7

Назва файлу:

Розмір файлу:

Початок діапазону:

Кінець діапазону:

Множник:

Error

Overflow!

Посилання на GitHub: <https://github.com/YuriiPatriuk12/Labs.git>