Автор: Наймитенко С. КІТ-119а

Дата: 09 квітня 2020

# **Лабораторна робота №9**

**ВИКЛЮЧЕННЯ**

Мета роботи: навчитись розробляти програми з реалізацією виключень

1 ЗАВДАННЯ ДО РОБОТИ

У файлі розміщена інформація про N масивів. У першому рядку міститься інформація про кількість масивів, у кожній наступній – інформація про кількість елементів у кожному масиві та власне дані масиву.

Необхідно реалізувати програму, що виконує перераховані нижче дії, причому кожна з них в окремій функції, поки користувач не введе замість назви файлу рядок \exit

Дії, що має виконувати програма, такі: - введення з клавіатури назви вхідного файлу з даними;

- читання даних з файлу;

- виконання індивідуального завдання;

- введення з клавіатури імені вихідного файлу;

- запис результату операції у файл;

- доступ до елемента за індексом слід винести в окрему функцію, що виконує перевірку на можливість виходу за межі масиву.

**Опис змінних**

Int\*\* array – масив елементів що були зчитані з файлу;

Int size\_s – кількість рядків в масиві;

Int\* size\_str – кількість елементів в рядку масиві;

String name\_file,write\_file – змінна назви файлу для зчитування та запису;

Int\* array\_max\_value – масив максимальних значень в кожному рядку масиву;

Int el – змінна для елементу, яку користувач хоче отримати;

**Опис функцій**

Bool Test\_Max\_value() – тест для функції Max\_value

Bool Test\_Return\_elements() - тест для функції Return\_elements

int\* Size\_array(string,int&) - визначення кількості рядків та елементів в них;

int\*\* Read\_File(string, int, int\*,int\*\*) – зчитування з файлу;

void Output\_array(int\*\*, int, int\*) – виведення масиву на екран;

int\* Max\_value(int\*\*, int, int\*) – визначення максимального значення в кожному рядку;

int Return\_elements(int\*\*, int, int\*,int,int) – повернення елементу по індексу;

void Write\_File(string, int, int\*,int\*\*) – запис до файлу;

**Текст програми**

Файл Function.h

#pragma once

#include"Header.h"

int\* Size\_array(string,int&);

int\*\* Read\_File(string, int, int\*,int\*\*);

void Output\_array(int\*\*, int, int\*);

int\* Max\_value(int\*\*, int, int\*);

int Return\_elements(int\*\*, int, int\*,int,int);

void Write\_File(string, int, int\*,int\*\*);

Файл Header.h

#pragma once

#define CRTDBG\_MAP\_ALLOC

#include <crtdbg.h>

#define DEBUG\_NEW new(\_NORMAL\_BLOCK, FILE, \_\_LINE)

#include <iostream>

#include <clocale>

#include <ctime>

#include <iomanip>

#include <fstream>

#include <sstream>

#include <regex>

#include <Windows.h>

#include <stddef.h>

#include <cstddef>

using std::cin;

using std::cout;

using std::string;

using std::endl;

using std::setw;

using std::ifstream;

using std::ofstream;

using std::stringstream;

using std::getline;

using std::istringstream;

using std::ostream;

using std::istream;

using std::exception;

Файл Function.cpp

#include "Header.h"

#include "Functions.h"

int\* Size\_array(string name\_file,int &size\_s)

{

ifstream file(name\_file);

if (!file)

throw exception("Внимание, файл не открылся!!!");

string line;

if (!getline(file, line))

throw "Внимание, файл пустой";

istringstream str(line);

str >> size\_s;

int\* size\_str = new int[size\_s];

for (int i = 0; i < size\_s; i++)

{

getline(file, line);

istringstream str(line);

str>>size\_str[i];

}

file.close();

return size\_str;

}

int\*\* Read\_File(string file\_name,int size\_s,int\* size\_str,int\*\* array)

{

ifstream file(file\_name);

if (!file)

throw 2.1;

string line;

int temp;

getline(file, line);

istringstream str(line);

str >> temp;

cout << "Происходит заполение масива данными" << endl;

for (int i = 0; getline(file, line); i++)

{

istringstream str(line);

str >> temp;

for (int j = 0; j < size\_str[i]; j++)

str >> array[i][j];

}

file.close();

return array;

}

void Output\_array(int\*\* array, int count\_s, int \*count\_str)

{

cout << "Происходит вывод на екран" << endl;

for (int i = 0; i < count\_s; i++)

{

for (int j = 0; j < count\_str[i]; j++)

cout << setw(3) << array[i][j];

cout << endl;

}

}

int\* Max\_value(int\*\* array, int count\_s, int\* count\_str)

{

int\* array\_max\_value = new int[count\_s];

cout << "Происходит поиск максимальных значений в масиве" << endl;

for (int i = 0; i < count\_s; i++)

{

array\_max\_value[i] = array[i][0];

for (int j = 0; j < count\_str[i]; j++)

if (array\_max\_value[i] < array[i][j])

array\_max\_value[i] = array[i][j];

}

return array\_max\_value;

}

int Return\_elements(int\*\* array, int count\_s, int\* count\_str, int ind\_i, int ind\_j)

{

if (ind\_i >= count\_s || ind\_i <= 0 || count\_str[ind\_i] <= ind\_j || ind\_j <= 0)

throw "Ошибка введения индексов";

int return\_el = 0;

cout << "Происходит поиск нужного елемента масива" << endl;

for (int i = 0; i < count\_s; i++)

{

for(int j=0;j<count\_str[i];j++)

if (i == ind\_i && j == ind\_j)

{

return\_el = array[i][j];

i = count\_s;

j = count\_str[i];

}

}

return return\_el;

}

void Write\_File(string file\_name, int count\_s, int\* count\_str,int\*\* array)

{

ofstream file(file\_name);

if (!file)

throw exception("Внимание, файл не открылся!!!");

cout << "Происходит заполение файла данными" << endl;

for (int i = 0; i < count\_s; i++)

{

for (int j = 0; j < count\_str[i]; j++)

file << array[i][j] << " ";

file << endl;

}

file.close();

}

Файл Test.cpp

#include "Header.h"

#include "Functions.h"

bool Test\_Max\_value()

{

int size\_s = 3;

int size\_str[3] = { 3,3,3 };

int\*\* array = new int\* [3];

for (int i = 0; i < 3; i++)

array[i] = new int[3];

int\* max\_value = new int[3];

for(int i=0;i<3;i++)

for (int j = 0; j < 3; j++)

{

if (i == 0)

{

array[i][j] = j;

max\_value[i] = j;

}

else if (i == 1)

{

array[i][j] = i + j;

max\_value[i] = i+j;

}

else

{

array[i][j] = 2 \* j;

max\_value[i] = 2 \* j;

}

}

int\* check = Max\_value(array,size\_s, size\_str);

int count\_true\_result = 0;

for (int i = 0; i < size\_s; i++)

if (max\_value[i] == check[i])

count\_true\_result++;

if (count\_true\_result == 3)

return true;

else

return false;

}

bool Test\_Return\_elements()

{

int size\_s = 3;

int size\_str[3] = { 3,3,3 };

int\*\* array = new int\* [3];

for (int i = 0; i < 3; i++)

array[i] = new int[3];

int\* max\_value = new int[3];

for (int i = 0; i < 3; i++)

for (int j = 0; j < 3; j++)

{

if (i == 0)

{

array[i][j] = j;

max\_value[i] = j;

}

else if (i == 1)

{

array[i][j] = i + j;

max\_value[i] = i + j;

}

else

{

array[i][j] = 2 \* j;

max\_value[i] = 2 \* j;

}

}

int check;

try {

check = Return\_elements(array, size\_s, size\_str, 2, 2);

}

catch (const char& ex)

{

cout << "ОШИБКА!!!" << ex << endl;

}

catch (...)

{

cout << "Хм, кажется непредвиденная ошибка в блоке try !!!" << endl;

for (int i = 0; i < size\_s; i++)

delete[] array[i];

delete[] array;

delete[] size\_str;

return 102;

}

int count\_true\_result = 0;

if (check == 4)

count\_true\_result++;

if (count\_true\_result == 1)

return true;

else

return false;

}

int main()

{

setlocale(LC\_ALL, "Rus");

bool result;

result = Test\_Max\_value();

if (result) /\*\*< перевірка тесту\*/

cout << "Test\_Max\_value пройден успешно!" << endl;

else

cout << "Test\_Max\_value не пройден!" << endl;

result = Test\_Return\_elements();

if (result) /\*\*< перевірка тесту\*/

cout << "Test\_Max\_value пройден успешно!" << endl;

else

cout << "Test\_Max\_value не пройден!" << endl;

}

Файл Main.cpp

#include "Header.h"

#include "Functions.h"

int main()

{

setlocale(LC\_ALL, "Rus");

string name\_file,write\_file;

int size\_s = 0, \*size\_str=0;

int\*\* array=0;

bool Flag = true;

while (Flag)

{

try

{

cout << "Введите название файла, с которого нужно считать данные" << endl;

cin >> name\_file;

if (name\_file == "\exit")

throw name\_file;

size\_str = Size\_array(name\_file,size\_s);

array = new int\* [size\_s];

for (int i = 0; i < size\_s; i++)

array[i] = new int[size\_str[i]];

array = Read\_File(name\_file, size\_s, size\_str,array);

if (size\_s == 0)

throw size\_s;

Flag = false;

}

catch (const exception & ex)

{

cout << ex.what() << endl << "Возможно вы ввели не верное название файла" << endl;

}

catch (const float ex)

{

cout << endl << "Что пошло не так, кажется файл стал не доступен!!! Завершение роботы." << endl;

return 1;

}

catch (const char\* ex)

{

cout << "ОШИБКА!!!" << endl << ex << endl;

Flag = false;

}

catch (int ex)

{

cout << "Переменные size\_s,size\_t oказались не заполнеными!!!" << endl;

return 1;

}

catch (string ex)

{

cout << "Введена команда завершение работы програмы!!!" << endl << ex << endl;

return 0;

}

catch (...)

{

cout << "Хм, кажется непредвиденная ошибка в 1м блоке try !!!" << endl;

return 102;

}

}

Output\_array(array, size\_s, size\_str);

int\* array\_max\_value = Max\_value(array, size\_s, size\_str);

cout << "Вывод максимальных значений каждого масива" << endl;

for (int i = 0; i < size\_s; i++)

cout << setw(3) << array\_max\_value[i];

Flag = true;

while (Flag)

{

try

{

int i, j;

cout<< "Введите индексы елемента (их 2) данные которого хотите получить" << endl;

cin >> i >> j;

int el = Return\_elements(array, size\_s, size\_str, i-1, j-1);

cout << "Елемент который вы искали : " << el << endl;

Flag = false;

}

catch (const char& ex)

{

cout << "ОШИБКА!!!"<< ex << endl;

}

catch (...)

{

cout << "Хм, кажется непредвиденная ошибка в 2м блоке try !!!" << endl;

for (int i = 0; i < size\_s; i++)

delete[] array[i];

delete[] array;

delete[] size\_str;

return 102;

}

}

Flag = true;

while (Flag)

{

try

{

cout << "Введите название файла, с которого нужно считать данные" << endl;

cin >> write\_file;

if (write\_file == "\exit")

throw write\_file;

Write\_File(write\_file, size\_s, size\_str,array);

Flag = false;

}

catch (const exception & ex)

{

cout << ex.what() << endl << "Возможно вы ошиблись с названием файла, попробуйте еще раз" << endl;

}

catch (string ex)

{

cout << "Введена команда завершение работы програмы!!!" << endl << ex << endl;

for (int i = 0; i < size\_s; i++)

delete[] array[i];

delete[] array;

delete[] size\_str;

return 0;

}

catch (...)

{

cout << "Хм, кажется непредвиденная ошибка в 3м блоке try !!!" << endl;

for (int i = 0; i < size\_s; i++)

delete[] array[i];

delete[] array;

delete[] size\_str;

return 103;

}

}

for (int i = 0; i < size\_s; i++)

delete[] array[i];

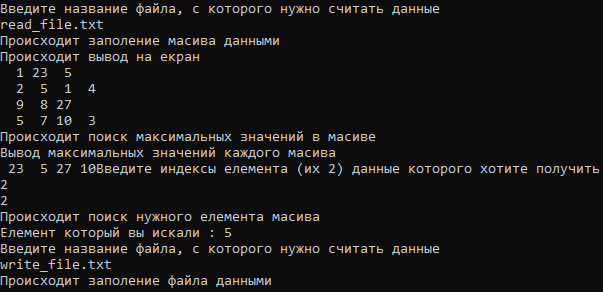
delete[] array;

delete[] size\_str;

return 0;

}

4 РЕЗУЛЬТАТ РОБОТИ ПРОГРАМИ



Результат роботи основоного завдання

C:\Users\fox\AppData\Local\Microsoft\Windows\INetCache\Content.Word\Screenshot_5.png

Результат тестування методів

ВИСНОВКИ

Навчились розробляти програми з реалізацією виключень. Виконали загальне завдання та виконали індивідуальне завдання згідно з варіантом 14 у списку.