

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Санкт-Петербургский государственный технологический институт (технический университет)»

Дисциплина: «Программирование»

Отчёт по лабораторной работе № 1

**Лабораторная работа №1. Использование массивов**

**Выполнил студент группы №485:**

**Проверили:**

Иван Григорьевич Корниенко

Алексей Константинович Федин

Санкт-Петербург

2019

# Постановка задачи

Задан массив А из N строк и M столбцов. Вывести на экран сумму значений всех элементов исходного массива. Заменить 1 элемент разностью между первым и вторым элементом, второй элемент разностью между вторым и третьим и т.д. Последний элемент остается неизменным. Вывести на экран сумму значений всех элементов полученного массива.

# Исходные данные

В качестве исходных данных программа использует вводимое пользователем количество строк и столбцов. В случае выбора пользователем заполнения массива из файла, программа запросит ввести путь к этому файлу, где первыми двумя значениями должны быть количества строк и столбцов, остальные же числа – элементы массива.

# Особые ситуации

* Если сумма строк и столбцов, указанных пользователем меньше двух, то программа запросит ввести их количество заново.
* Если пользователь при указании пути к файлу будет использовать запрещённые имена, например: con, aux и т.д., то программа попросит ввести путь к файлу заново.
* Если в файле, из которого должен быть заполнен массив, вместо числа будет найден символ или в конце файла будут пробелы, то программа попросит пользователя исправить файл и ввести путь заново.

# Математические методы и алгоритмы решения задач

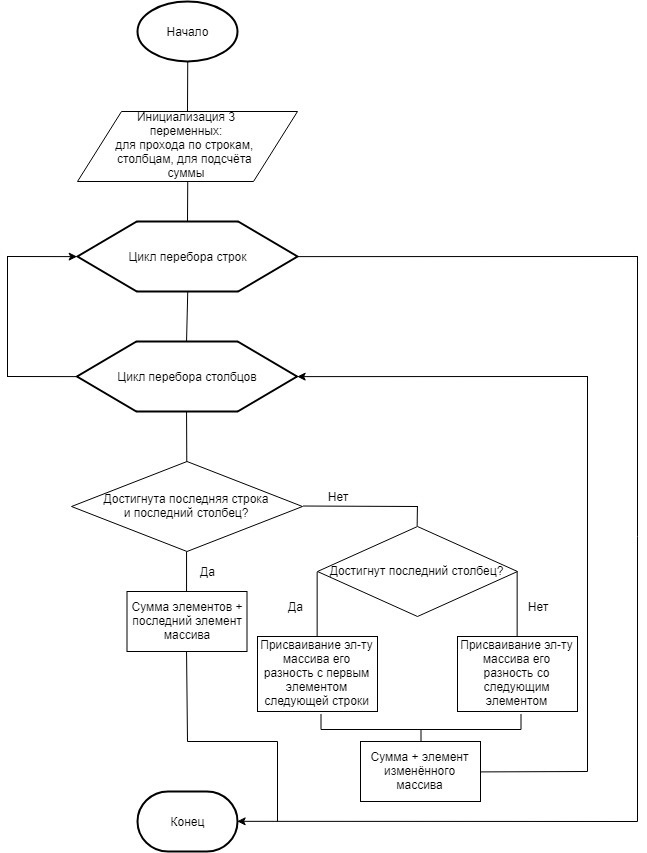


Рисунок 1 – Блок-схема подсчёта суммы изначального массива, замены каждого элемента массива, кроме последнего, на его разность с последующим, а также подсчёта суммы полученного массива

Согласно постановке задачи, для составления программы будут использоваться алгоритмы, блок схема которых представлена выше.

# Форматы представления данных

Программа использует следующие переменные:

Таблица 1 – Переменные, используемы в программе

|  |  |  |
| --- | --- | --- |
| **Имя** | **Тип** | **Описание** |
| lines | unsigned int | Количество строк массива |
| columns | unsigned int | Количество столбцов массива |
| running | bool | Определяет, запущена ли программа |
| menu | unsigned int | Для ввода пункта меню |
| i | int | Для цикла чередования строк |
| j | int | Для цикла чередования столбцов |
| fileArr | ifstream | Занесение значений из файла в массив |
| path | string | Адрес файла |
| count | int | Проверка количества строк и столбцов в файле |
| checkValue | int | Проверка количества строк и столбцов в файле |
| checkSymbol | char | Проверка на числа, вводимые из файла |
| arr | int\*\* | Массив |
| choice | bool | Хочет ли пользователь записать массив в файл |
| sourceArrSum | int | Сумма элементов исходного массива |
| changedArrSum | int | Сумма элементов изменённого массива |
| color | int | Цвет вывода в консоль |
| fileOutput | ofstream | Вывод данных в файл |

Для задания максимального и минимального размера массивов, предела генерирования случайных чисел, а также ввода максимального пункта меню используются следующие константы:

Таблица 2 – Константы, используемы в программе

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Тип** | **Значение** | **Описание** |
| INT\_MIN | const int | -2147483647 | Минимальное целое число |
| INT\_MAX | const int | 2147483647 | Максимальное целое число |
| maxMenuNumber | const int | 3 | Максимальный номер пункта меню |

# Структура программы

В силу большого количества функций программа разделена на семь исполняемых модулей, из которых один является основным и отвечает за запуск программы, пять оставшихся содержат в себе функции, необходимые для работы программы. Последний файл отвечает за предкомпиляцию и сокращает время сборки программы с 5 до 2-х секунд.

Модуль Menu:

Таблица 3 – Функции, составляющие модуль Menu

|  |  |
| --- | --- |
| **Имя** | **Описание** |
| MainMenu | Вывод главного меню, выбор пункта меню |

Модуль ArrFilling:

Таблица 4 – Функции, составляющие модуль ArrFilling

|  |  |
| --- | --- |
| **Имя** | **Описание** |
| FileInput | Ввод элементов из файла в массив |
| ManualInput | Ввод элементов массива из консоли вручную |
| RandomFilling | Заполнение массива случайными элементами |

Модуль ArrChange:

Таблица 5 – Функции, составляющие модуль ArrChange

|  |  |
| --- | --- |
| **Имя** | **Описание** |
| ArrChange | Изменение массива, подсчёт сумм |

Модуль Output:

Таблица 6 – Функции, составляющие модуль Output

|  |  |
| --- | --- |
| **Имя** | **Описание** |
| SetColor | Изменение цвета текста, выводимого в консоль |
| OutputArrInFile | Вывод исходного массива в файл |
| OutputResultInFile | Вывод результата в файл |
| OutputOrigArr | Проверка на корректность введенного пути файла |
| OutputResult | Проверка на корректность введенного пути файла |
| OutputInConsole | Вывод массива в консоль |

Модуль InputAndCheck:

Таблица 7 – Функции, составляющие модуль InputAndCheck

|  |  |
| --- | --- |
| **Имя** | **Описание** |
| GetInput | Проверка на правильность введённых данных задаваемого типа |
| GetInt | Проверка на правильность введённых данных типа int |
| GetUnsignedInt | Проверка на правильность введённых данных типа unsigned int |
| GetBool | Проверка на правильность введённых данных типа bool |

# Описание хода выполнения лабораторной работы

* В ходе лабораторной работы было создано решение (Lab1) в интегрированной среде разработки Microsoft Visual Studio C++ 2017. В нём был создан проект.
* В созданном проекте нужно было включить все библиотеки и заголовочные файлы, использованные в программе, в предкомпилированный заголовок pch.h для её более быстрого запуска
* При работе программы с файлами нужно было добавить проверки на валидность имени файла, а также на то, создан ли файл или нет при сохранении.
* При получении пользовательского ввода необходимо было добавить проверку, на его соответствие предполагаемому типу данных и условиям выбора.
* Перед повторением программы необходимо очищать память для того, чтобы не возникало непредвиденных ошибок, связанных с заполнением областей памяти старыми числами.
* Программа после запуска выдавала одни и те же результаты, хотя в коде использовался вызов функции rand, возвращающей случайное число. После изучения справочной системы выяснилось, что необходимо использовать функцию srand для начальной инициализации генератора случайных чисел. После этого программа стала работать правильно.

# Результат работы программы

В результате работы программа выводит два массива различных цветов, первый массив является исходным, а второй — результатом работы программы

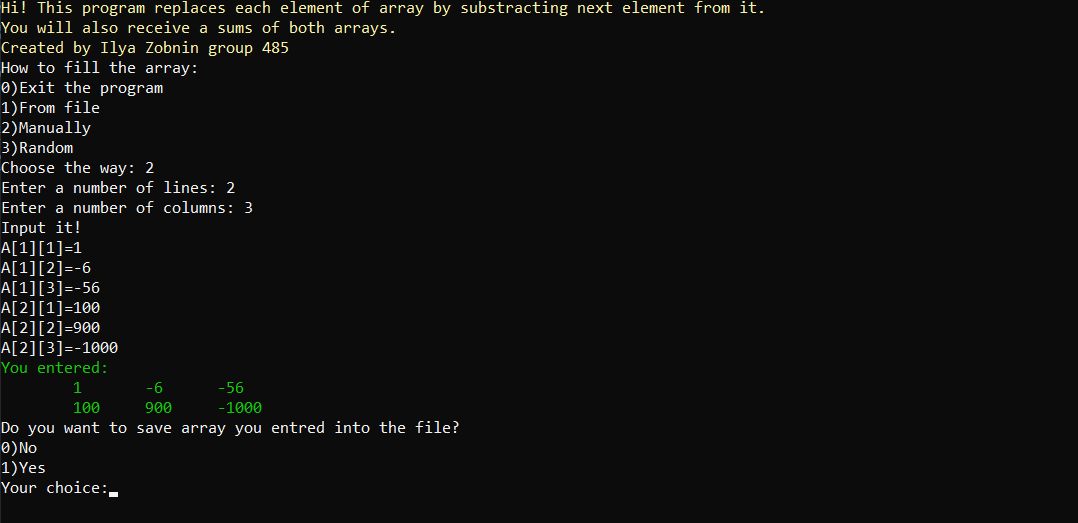


Рисунок 2 – Заполнение массива с клавиатуры

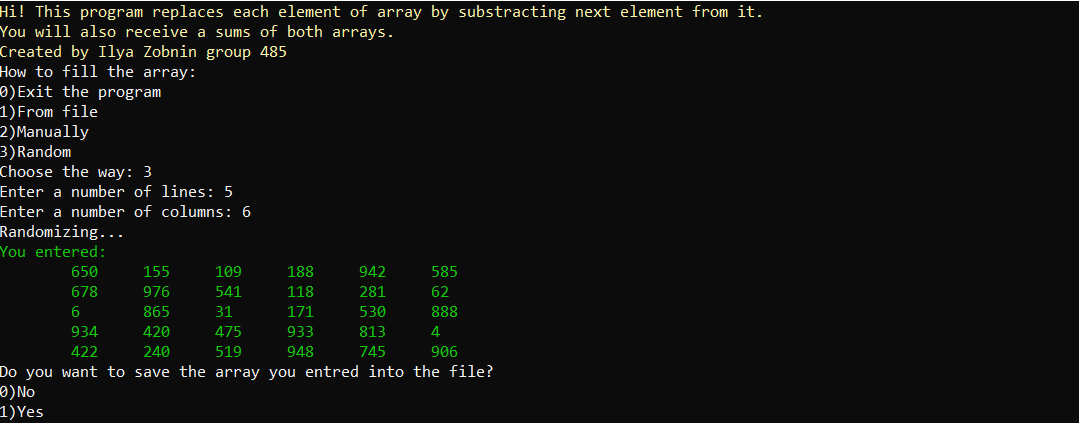


Рисунок 3 – Заполнение массива случайными числами

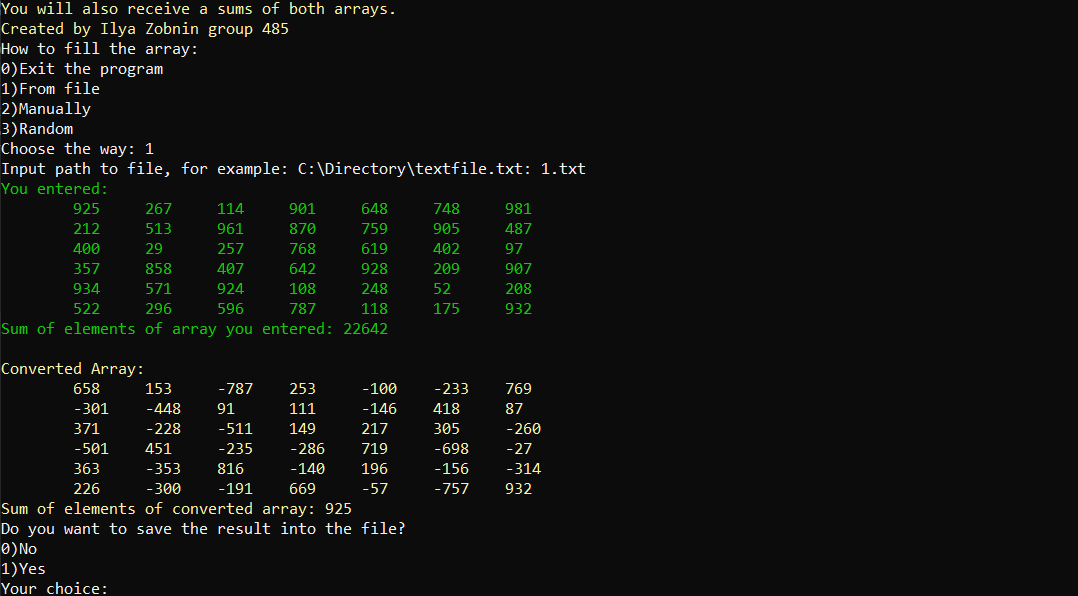


Рисунок 4 – Заполнение массива из файла

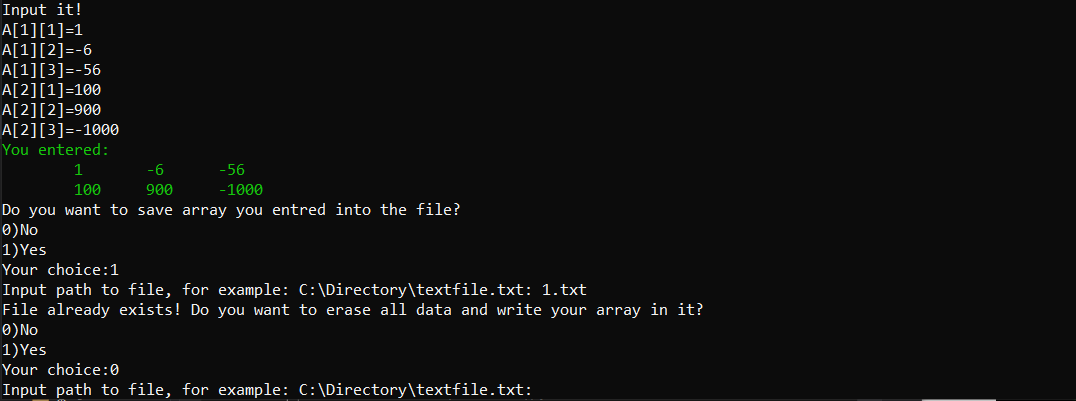


Рисунок 5 – Попытка сохранить в уже созданный файл

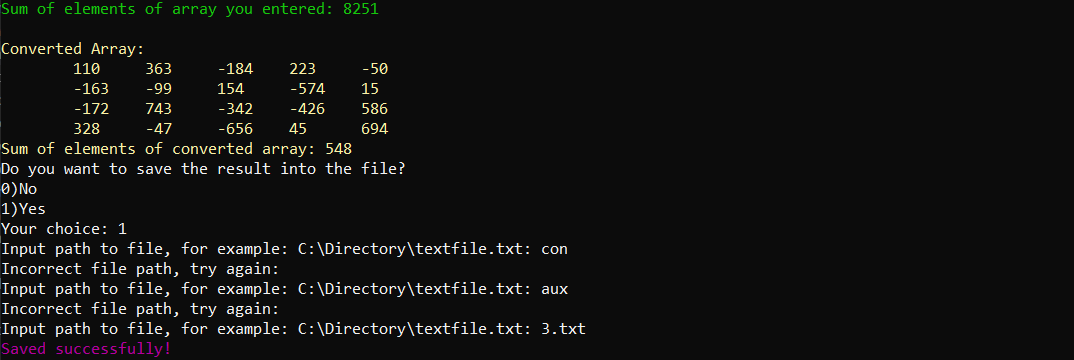


Рисунок 6 – Попытка сохранить в файл с запрещенным именем

# Текст программы

# [--- Начало программы ---]

**// pch.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

**// pch.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

#pragma once

#include <iostream>

#include <Windows.h>

#include <experimental\filesystem>

#include <fstream>

**// ArrFilling.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

void FileInput();

void RandomFilling();

void ManualInput();

**// Menu.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

bool MainMenu();

**// ArrChange.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

void ArrChange(int \*\*arr, int lines, int columns);

**// Output.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

void OutputInConsole(int \*\*arr, int lines, int columns);

void SetColor(int color);

void OutputOrigArr(int \*\*arr, int lines, int columns);

void OutputResult(int \*\*arr, int lines, int columns, int sourceArrSum, int changedArrSum);

**// InputAndCheck.h**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#pragma once

enum {green = 10, purple = 13, yellow, white};

int GetInt();

int GetUnsignedInt();

bool GetBool();

**// Lab 1.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

#include "Output.h"

#include "InputAndCheck.h"

#include "Menu.h"

using namespace std;

int main() {

bool running = 1;

SetColor(yellow);

cout << "Hi! This program replaces each element of array by substracting next element from it." << endl <<

"You will also receive a sums of both arrays." << endl << "Created by Ilya Zobnin group 485" << endl;

SetColor(white);

while (running)

running = MainMenu();//переход к функции, демонстрирующей меню

return 0;

}

**// Menu.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

#include "InputAndCheck.h"

#include "ArrFilling.h"

using namespace std;

enum { close, fromFile, manually, random };

bool MainMenu() {

bool running = true;

cout << "How to fill the array: " << endl << "0)Exit the program" << endl << "1)From file" << endl << "2)Manually" << endl << "3)Random" << endl;

cout << "Choose the way:”;

int menu;

const int maxMenuNumber = 3;

menu = GetUnsignedInt();

if (menu > maxMenuNumber) {

cout << "Try again: " << endl;

MainMenu();

}

else

switch (menu) {

case fromFile: {

FileInput();

running = true;

break;

}

case manually: {

ManualInput();

running = true;

break;

}

case random: {

RandomFilling();

running = true;

break;

}

case close: {

running = close;

break;

}

}

return running;

}

**// ArrFilling.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

#include "InputAndCheck.h"

#include "Output.h"

#include "ArrChange.h"

using namespace std;

using namespace experimental::filesystem;

void FileInput() {//получение значений из файла

int lines, columns, i, j;

string path;

cout << "Input path to file, for example: C:\\Directory\\textfile.txt: ";

cin >> path;

cin.ignore(INT\_MAX, '\n');

ifstream filearr;

filearr.open(path);

if (!is\_regular\_file(path)) {

cout << "Incorrect file path, try again: " << endl;

filearr.close();

FileInput();

}

else {

int count = -2, check;

while (!filearr.eof()) {

filearr >> check;

count++;

if (filearr.fail()) {

filearr.close();

cout << "Invalid value type or there is an extra space in the end of the file. Edit the file and try again!" << endl;

FileInput();

break;

}

}

filearr.close();

filearr.open(path);

filearr >> lines;

filearr >> columns;//ввод кол-ва строк и столбцов, как первые два значения в файле

if ((lines + columns) < 2) {

cout << "Lines and columns values must be more than 1. Edit the file and try again!" << endl;

filearr.close();

FileInput();

}

else{

if (!(count == lines \* columns)) {

cout << "Not enough elements in file, edit the file and try again!" << endl;

filearr.close();

FileInput();

}

else {

int \*\*arr = new int\*[lines];

for (i = 0; i < lines; i++)

arr[i] = new int[columns];//создание динамического массива

for (i = 0; i < lines; i++)

for (j = 0; j < columns; j++)

filearr >> arr[i][j];//занесение значений из файла в массив

SetColor(green);

cout << "You entered: ";

OutputInConsole(arr, lines, columns);//вывод изначального массива в консоль

SetColor(white);

ArrChange(arr, lines, columns);

//выполнение задания по изменению массива и вывода сумм

}

}

}

}

void RandomFilling() {//случайное заполнение

srand(static\_cast <unsigned int> (time(nullptr)));

//для генерации случайных чисел при каждом запуске

int lines = 0, columns = 0;

bool choice;

cout << "Enter a number of lines: ";

lines = GetUnsignedInt();

cout << "Enter a number of columns: ";

columns = GetUnsignedInt();//запрос кол-ва строк и стобцов от пользователя

if ((lines + columns) > 2) {

int \*\*arr = new int\*[lines];

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

arr[i] = new int[columns];

cout << "Randomizing..." << endl;

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

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

arr[i][j] = rand() % 1001;

SetColor(green);

cout << "You entered: ";

OutputInConsole(arr, lines, columns);//вывод изначального массива в консоль

SetColor(white);

cout << endl << "Do you want to save the array you entred into the file?" << endl

<< "0)No" << endl

<< "1)Yes" << endl

<< "Your choice:";

choice = GetBool();

//запрос от пользователя, хочет ли он сохранить полученный массив в файл

if (choice)

OutputOrigArr(arr, lines, columns);

ArrChange(arr, lines, columns);//выполнение задания по изменению массива и вывода сумм

}

else {

cout << "Invalid lines and columns values, try again!" << endl;

RandomFilling();

}

}

void ManualInput() {//ввод всех данных вручную

int lines, columns;

bool choice;

cout << "Enter a number of lines: ";

lines = GetUnsignedInt();

cout << "Enter a number of columns: ";

columns = GetUnsignedInt();

if ((lines + columns) > 2) {

int \*\*arr = new int\*[lines];

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

arr[i] = new int[columns];

cout << "Input it!" << endl;

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

for (int j = 0; j < columns; j++) {

cout << "A[" << i + 1 << "]"

<< "[" << j + 1 << "]=";

arr[i][j] = GetInt();//ввод значений вручную

}

SetColor(green);

cout << "You entered: ";

OutputInConsole(arr, lines, columns);//вывод изначального массива в консоль

SetColor(white);

cout << endl << "Do you want to save array you entred into the file?" << endl

<< "0)No" << endl

<< "1)Yes" << endl

<< "Your choice:";

choice = GetBool();

//запрос от пользователя, хочет ли он сохранить полученный массив в файл

if (choice)

OutputOrigArr(arr, lines, columns);

ArrChange(arr, lines, columns);

//выполнение задания по изменению массива и вывода сумм

}

else {

cout << "Invalid lines and columns values, try again!" << endl;

ManualInput();

}

}

**// ArrChange.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

#include "InputAndCheck.h"

#include "Output.h"

using namespace std;

void ArrChange(int \*\*arr, int lines, int columns) {

int sourceArrSum = 0, changedArrSum = 0, i, j;

for (i = 0; i < lines; i++)

for (j = 0; j < columns; j++)

sourceArrSum += arr[i][j];//подсчёт суммы эелементов изначального массива

for (i = 0; i < lines; i++)

for (j = 0; j < columns; j++) {

if (i == lines - 1 && j == columns - 1) {//проверка на последний элемент массива

changedArrSum += arr[i][j];

break;//остановка цикла, не изменение последнего элемента массива

}

if (j == columns - 1)//если последний столбец - вычитание первого элемента следующей строки

arr[i][j] -= arr[i + 1][0];

else

arr[i][j] -= arr[i][j + 1];

changedArrSum += arr[i][j];

}

//изменение цветов, вывод изменённого массива и сумм элементов начального и изменённого массивов

SetColor(green);

cout << endl << "Sum of elements of array you entered: " << sourceArrSum;

SetColor(yellow);

cout << endl << endl << "Converted Array: ";

OutputInConsole(arr, lines, columns);

cout << endl << "Sum of elements of converted array: " << changedArrSum;

SetColor(white);

cout << endl << "Do you want to save the result into the file?" << endl << "0)No" << endl << "1)Yes" << endl << "Your choice: ";

bool choice;

choice = GetBool();//хочет ли пользователь сохранить результат в файл

if (choice)

OutputResult(arr, lines, columns, sourceArrSum, changedArrSum);//сохранение в файл

//полная очистка массива и его указателя

for (i = 0; i < lines; i++) {

delete[] arr[i];

arr[i] = nullptr;

}

delete[] arr;

arr = nullptr;

}

**// InputAndCheck.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

using namespace std;

template <typename T>//использование шаблона для использования вариабельного типа данных

T GetInput() {

T userInput;

cin >> userInput;//ввод пользователем необходимых данных

while (cin.fail()) {//цикл пока ввод данных не соответствует заданному типу

cout << "Try again: " << endl;

cin.clear();//обнуление cin.fail

cin.ignore(INT\_MAX, '\n');//игнорирование введённых данных

cin >> userInput;//повторный ввод переменной

}

cin.ignore(INT\_MAX, '\n');

return userInput;

}

//все последующие функции используют предыдущую для ввода пользователем заданных типов данных

int GetInt() {

return GetInput<int>();

}

int GetUnsignedInt() {

int i = GetInput<int>();

if (i < 0) {

cout << "Try again: ";

return GetUnsignedInt();

}

else

return i;

}

bool GetBool(){

return GetInput<bool>();

}

**// Output.cpp**

**// Лабораторная работа №1.**

**// Студент группы 485, Зобнин Илья Михайлович. 2019 год**

#include "pch.h"

#include "InputAndCheck.h"

using namespace std;

using namespace experimental::filesystem;

void SetColor(int color) {//функция для изменения цвета текста в консоли

SetConsoleTextAttribute(GetStdHandle(STD\_OUTPUT\_HANDLE), color);

}

void OutputArrInFile(int \*\*arr, int lines, int columns, string path) {

//функция для вывода кол-ва строк, столбцов и изначального массива

ofstream fileOutput;

fileOutput.open(path);

fileOutput << lines << endl;

fileOutput << columns;

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

fileOutput << endl;

for (int j = 0; j < columns; j++) {

if (j == columns - 1)

fileOutput << arr[i][j];

else

fileOutput << arr[i][j] << " ";

}

}

fileOutput.close();

SetColor(purple);

cout << "Saved successfully! " << endl;

SetColor(white);

}

void OutputResultInFile(int \*\*arr, int lines, int columns, string path, int sourceArrSum, int changedArrSum) {

//функция для вывода изменённого массива, сумм элементов начального и изменённого массивов

ofstream fileOutput;

fileOutput.open(path);

fileOutput << "Sum of elements of array you entered: " << endl;

fileOutput << sourceArrSum << endl;

fileOutput << "Converted Array: ";

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

fileOutput << endl;

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

fileOutput << arr[i][j] << " ";

}

fileOutput << endl << "Sum of elements of converted array: " << endl;

fileOutput << changedArrSum << endl;

fileOutput.close();

SetColor(purple);

cout << "Saved successfully! " << endl;

SetColor(white);

}

void OutputInConsole(int \*\*arr, int lines, int columns) {

//функция для вывода массива в консоль

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

cout << endl;

for (int j = 0; j < columns; j++) {

cout << "\t" << arr[i][j] << " ";

}

}

}

void OutputOrigArr(int \*\*arr, int lines, int columns) {

//функция для проверки путей к файлу и вывода изначального массива

bool choice;

string path;

cout << "Input path to file, for example: C:\\Directory\\textfile.txt: ";

cin >> path;

cin.ignore(INT\_MAX, '\n');

if (is\_regular\_file(path)) {//если пользователь ввёл путь к уже созданному файлу

cout << "File already exists! Do you want to erase all data and write your array in it?" << endl << "0)No" << endl << "1)Yes" << endl << "Your choice:";

choice = GetBool();//пользователь выбирает, хочет ли он стереть все данные в файле и записать туда новые, либо указать путь снова

if (choice)//запись в уже созданный файл

OutputArrInFile(arr, lines, columns, path);

else//ввод пути заного

OutputOrigArr(arr, lines, columns);

}

else {

ofstream fileCheck;

fileCheck.open(path);

if (is\_regular\_file(path)) {

fileCheck.close();

OutputArrInFile(arr, lines, columns, path);

}

else {//если такого файла не существует в системе

fileCheck.close();

cout << "Incorrect file path, try again: " << endl;

OutputOrigArr(arr, lines, columns);//возвращение к начальной точке

}

}

}

void OutputResult(int \*\*arr, int lines, int columns, int sourceArrSum, int changedArrSum) {

//функция для проверки путей к файлу, вывода изменённого массива, сумм элементов начального и изменённого массивов

string path;

bool choice;

cout << "Input path to file, for example: C:\\Directory\\textfile.txt: ";

cin >> path;

cin.ignore(INT\_MAX, '\n');

if (is\_regular\_file(path)) {

cout << "File already exists! Do you want to erase all data and write your array in it?" << endl << "0)No" << endl << "1)Yes" << endl << "Your choice:";

choice = GetBool();

//пользователь выбирает, хочет ли он стереть все данные в файле и записать туда новые, либо указать путь снова

if (choice)//запись в уже созданный файл

OutputResultInFile(arr, lines, columns, path, sourceArrSum, changedArrSum);

else//ввод пути заного

OutputResult(arr, lines, columns, sourceArrSum, changedArrSum);

}

else {//если такого файла не существует в системе

ofstream fileCheck;

fileCheck.open(path);

if (is\_regular\_file(path)) {

fileCheck.close();

OutputResultInFile(arr, lines, columns, path, sourceArrSum, changedArrSum);

}

else {

fileCheck.close();

cout << "Incorrect file path, try again: " << endl;

OutputResult(arr, lines, columns, sourceArrSum, changedArrSum);

}

}

}

**[--- Конец программы ---]**