Министерство науки и высшего образования Российской Федерации

Чайковский филиал

федерального государственного автономного образовательного учреждения

высшего образования

**«Пермский национальный исследовательский**

**политехнический университет»**

Кафедра Автоматизации, информационных и инженерных технологий

Направление: 09.03.01 Информатика и вычислительная техника

Направленность: Автоматизированные системы обработки информации и управления

**О Т Ч Е Т**

**по лабораторной работе №1**

«Функции и массивы»

по дисциплине

«Теория алгоритмов и структуры данных»

|  |  |  |
| --- | --- | --- |
|  |  | Выполнили: |
|  |  | студенты группы АСУ-22-1б ЧФ |
|  |  | *Базуев Д.О., Гимадеев Р.Р.* |
|  |  | *(Фамилия Имя Отчество)* |
|  |  |  |
|  |  | Проверил: |
|  |  | *Ст. преподаватель* |
|  |  | *(должность, ученая степень, ученое звание)* |
|  |  | *Сухих И.И.* |
|  |  | *(Фамилия Имя Отчество)* |
|  |  | *(дата, подпись)* |

Чайковский 2023

Цель работы:

1. Получение практических навыков при работе с одномерными, многомерными и рваными массивами.
2. Получение практических навыков при работе с функциями, передаче данных в функции различными способами, получение результатов из функций различными способами.
3. Получение практических навыков при создании диалоговых консольных приложений.

Программное обеспечение: Microsoft Visual Studio 2022.

Задание. Сформировать динамический одномерный массив, заполнить его числами и вывести на печать. Удалить первый отрицательный элемент и вывести полученный массив на печать. Сформировать динамический двумерный массив, заполнить его числами и вывести на печать. Добавить столбец с заданным номером и вывести полученный массив на печать. Сформировать динамический рваный массив, заполнить его случайными числами и вывести на печать. Удалить самую короткую строку.

Исходный код:

#include <iostream>

#include <string>

#include <windows.h>

#include <time.h>

#include <vector>

using namespace std;

int InputValidation(int num, string s);

int PositiveInputValidation(int num, string s);

int main()

{

srand(time(0));

setlocale(0, "");

bool b = false;

int choice1 = 0, choice2 = 0, n = 1, m = 1, DobNum = 0, ArrLenght = 0;

int\* mas1 = new int[1];

int\*\* mas2 = new int\* [1];

int\* mas4 = new int[1];

int\*\* mas3 = new int\* [1];

int k, min;

do

{

choice1 = 0;

// Меню

cout << "Список действий с массивами:\n"

"1. Одномерный массив. Удаление первого отрицательного элемента.\n"

"2. Двумерный массив. Добавление столбца с заданным номером.\n"

"3. Рваный массив. Удаление самой короткой строки.\n"

"4. Выход из программы\n";

choice1 = InputValidation(choice1, "\nВыберите действие: ");

switch (choice1)

{

case 1:

// Одномерный массив. Добавление элемента в начало массива

do

{

b = true;

n = PositiveInputValidation(n, "\nВведите количество чисел в одномерном массиве: ");

// n = IntInputValidation("\nВведите количество чисел в одномерном массиве: ");

mas1 = new int[n];

cout << "\nВыберите тип заполнения массива:\n"

"1. Вручную.\n"

"2. Рандомно.\n";

choice2 = InputValidation(choice2, "\nВаш выбор: ");

if (choice2 == 1)

{

// Заполнение вручную

cout << "\nЗаполнение массива целыми числами:\n";

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

{

mas1[i] = InputValidation(mas1[i], "Введите элемент массива: ");

}

// Вывод одномерного массива

cout << "\nИзначальный массив:\n";

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

{

cout << mas1[i] << " ";

}

b = false;

}

else

{

if (choice2 == 2)

{

// Автоматическое заполнение

cout << "\nПолученный одномерный массив:\n";

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

{

mas1[i] = rand() % 20 - 10;

cout << mas1[i] << " ";

}

b = false;

}

else

{

cout << "\nОшибка. Неверный тип заполнения.\n";

b = true;

}

}

} while (b);

// Удаление первого отрицательного элемента

k = 0;

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

{

if (k == 0)

{

if (mas1[i] < 0)

{

k += 1;

while (i != n)

{

mas1[i] = mas1[i + 1];

i++;

}

n -= 1;

}

}

}

// Вывод массива

cout << "\nМассив после удаления элемента:\n";

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

{

cout << mas1[i] << " ";

}

cout << "\n\n";

break;

case 2:

// Двумерный массив. Добавление столбца с заданным номером

do

{

b = true;

n = PositiveInputValidation(n, "\nВведите количество строк в массиве: ");

m = PositiveInputValidation(m, "\nВведите количество столбцов в массиве: ");

// Создание чистого двумерного массива

mas2 = new int\* [n];

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

{

mas2[i] = new int[m];

}

cout << "\nВыберите тип заполнения массива:\n"

"1. Вручную.\n"

"2. Рандомно.\n";

choice2 = InputValidation(choice2, "\nВаш выбор: ");

switch (choice2)

{

case 1:

// Заполнение вручную

cout << "\nЗаполнение массива целыми числами:\n";

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

{

cout << "Заполнение" << i + 1 << "-й строки целыми числами:\n";

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

{

mas2[i][j] = InputValidation(mas2[i][j], "Введите элемент массива: ");

}

}

b = false;

break;

case 2:

// Автоматическое заполнение

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

{

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

{

mas2[i][j] = rand() % 10;

}

}

b = false;

break;

default:

cout << "\nОшибка. Неверный тип заполнения.\n";

b = true;

break;

}

} while (b);

// Вывод двумерного массива

cout << "\nИзначальный массив:\n";

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

{

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

{

cout << mas2[i][j] << " ";

}

cout << "\n";

}

// Добавление столбца

do

{

b = false;

DobNum = PositiveInputValidation(DobNum, "\nВыберите столбец для добавления: ");

if (DobNum > m)

{

cout << "\nОшибка. Некорректная строка\n";

b = true;

}

} while (b);

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

{

mas2[i][m] = mas2[i][DobNum-1];

}

m++;

// Вывод обновленного двумерного массива

cout << "\nОбновленный двумерный массив:\n";

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

{

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

{

cout << mas2[i][j] << " ";

}

cout << "\n";

}

cout << "\n";

break;

case 3:

// Рваный массив. Добавление строк

cout << "Создание рваного массива: \n";

n = PositiveInputValidation(n, "\nВведите количество строк: ");

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

{

m = PositiveInputValidation(m, "\nВведите сколько элементов должно быть в строке: ");

mas4[i] = m;

}

do

{

b = true;

mas3 = new int\* [n];

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

{

mas3[i] = new int[mas4[i]];

}

cout << "\nВыберите тип заполнения массива:\n"

"1. Вручную.\n"

"2. Рандомно.\n";

choice2 = InputValidation(choice2, "\nВаш выбор: ");

switch (choice2)

{

case 1:

cout << "\nЗаполнение массива целыми числами:\n";

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

{

cout << "Заполнение " << i + 1 << "-й строки целыми числами"

", которая состоит из " << mas4[i] << "Элементов:\n";

for (int j = 0; j < mas4[i]; j++)

{

mas3[i][j] = InputValidation(mas3[i][j], "Введите элемент строки: ");

}

}

b = false;

break;

case 2:

// Заполнение массива рандомными числами

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

{

for (int j = 0; j < mas4[i]; j++)

{

mas3[i][j] = rand() % 11 - 5;

}

}

b = false;

break;

default:

cout << "\nОшибка. Неверный тип заполнения.\n";

b = true;

break;

}

} while (b);

// Вывод рваного массива

cout << "\nИзначальный массив:\n";

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

{

for (int j = 0; j < mas4[i]; j++)

{

cout << mas3[i][j] << " ";

}

cout << "\n";

}

// Удаление самой короткой строки

min = 100;

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

{

if (mas4[i] < min)

{

min = mas4[i];

}

}

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

{

if (min == mas4[i])

{

while (i <= n)

{

for (int j = 0; j < mas4[i+1]; j++)

{

while (mas4[i] < mas4[i+1])

{

mas4[i]++;

}

while (mas4[i] > mas4[i + 1])

{

mas4[i]--;

}

mas3[i] = mas3[i + 1];

}

i++;

}

n--;

}

}

cout << "\n";

// Вывод рваного массива

cout << "\nНовый массив:\n";

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

{

for (int j = 0; j < mas4[i]; j++)

{

cout << mas3[i][j] << " ";

}

cout << "\n";

}

break;

case 4:

break;

default:

cout << "Ошибка. Неверное действие.\n";

break;

}

} while (choice1 != 4);

return 0;

}

int InputValidation(int num, string s)

{

bool b = false;

do {

b = false;

cout << s;

if (!(cin >> num)) {

cin.clear();

cin.ignore((numeric\_limits<streamsize>::max)(), '\n');

cout << "Ошибка ввода!\n";

b = true;

}

} while (b);

return num;

}

int PositiveInputValidation(int num, string s)

{

do {

num = InputValidation(num, s);

if (num < 1)

{

cout << "Ошибка. Введено отрицательно число или 0\n";

}

} while (num < 1);

return num;

}

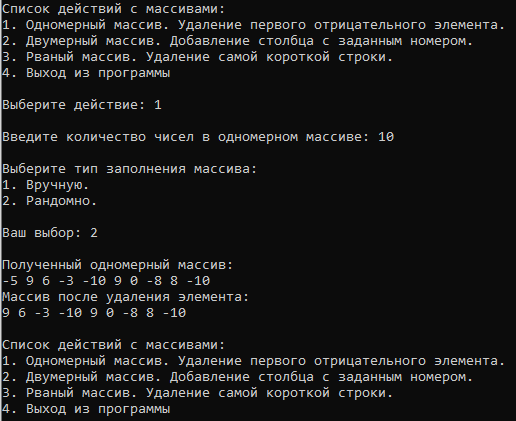


Рисунок 1 – Результат выполнения 1 пункта

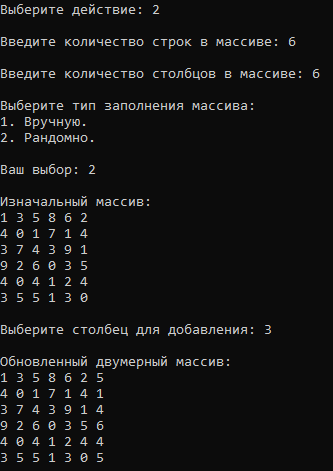
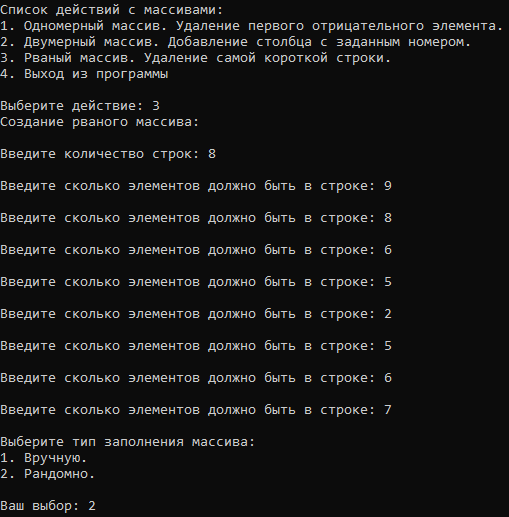


Рисунок 2 – Результат выполнения 2 пункта



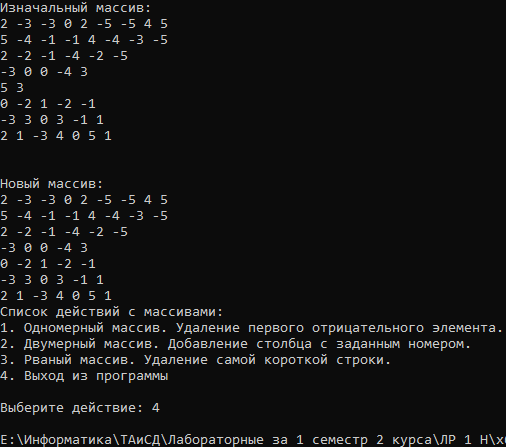


Рисунок 3 – Результат выполнения 3 и 4 пункта

Вывод: Были получены практические навыки при работе с одномерными, многомерными и рваными массивами, функциями, передаче данных в функции различными способами, получении результатов из функции различными способами, при создании диалоговых консольных приложений.