**ФГБОУ ВО**

**Национальный исследовательский университет**

**«МЭИ»**

**Лабораторная работа №2**

**Динамические структуры данных**

**Вариант 8**

**Задание №3:**

**Динамические массивы**

**Выполнил:**

студент группы Аэ-21-23

Нестеров А.С.

**Москва, 2024**

1. **Постановка задачи**

Задание:

Ввести массив А из n элементов. Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались в конце массива.

Функции программы:

1. создание массива из n элементов
2. Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались в конце массива.
3. Вывод массива

Входные данные:

* n(int) – количество элементов массива [1;10]
* Элементы массива (int) – ограничение (4 байта)

[-2147483648; 2147483647]

1. **Разработка приложения** 
   1. Разработка структуры приложения

Задание реализовано в виде консольного приложения на языке С++ с использованием процедур и функций.

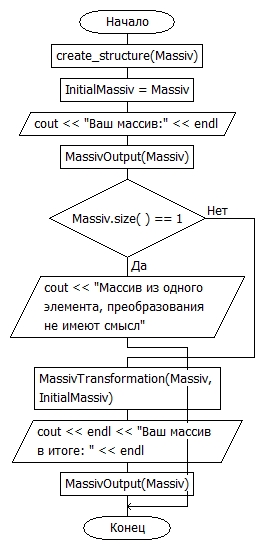


Рис. 1 – Структура приложения

Таблица 1 - Спецификация функций

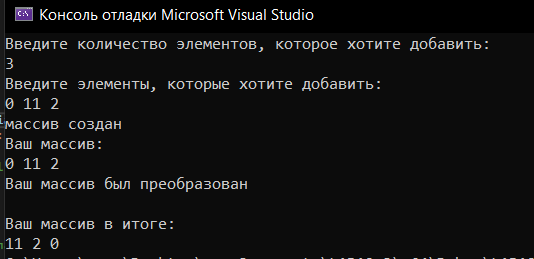
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Название | Назначение | Вх. данные | Вых. данные |
| 1 | create\_structure | создание массива из n элементов | Массив А  (количество элементов и сами элементы функция запрашивает сама) | Массив А |
| 2 | MassivTransformation | Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались в конце массива. | Массив А(который мы преобразовываем) и первоначальный массив, введенный пользователем (для сравнения) | Массив А |
| 3 | MassivOutput | Вывод массива | Массив А | Массив А |

2.3 Разработка пользовательского интерфейса

Категория пользователей: Студенты

Интерфейс представляет собой консольное приложение, в котором взаимодействие пользователя с программой осуществляется с помощью диалогового окна с выбором опций.

Пример диалогов с пользователем:



1. **Реализация и тестирование приложения**

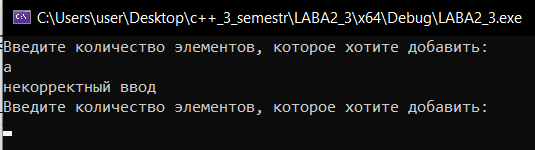
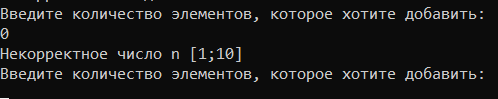
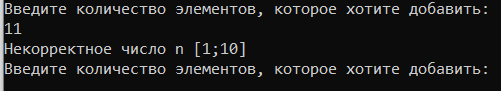
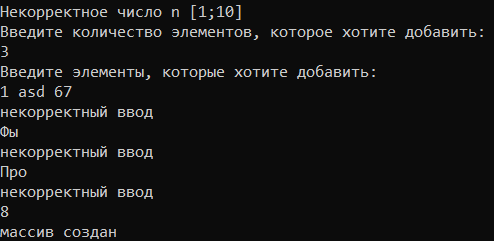
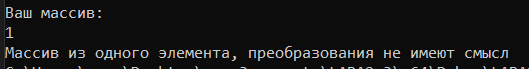
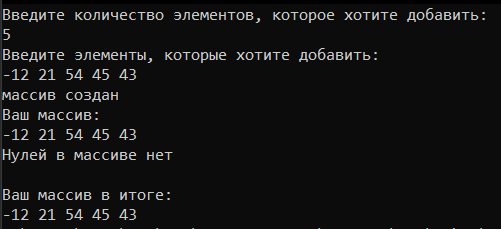
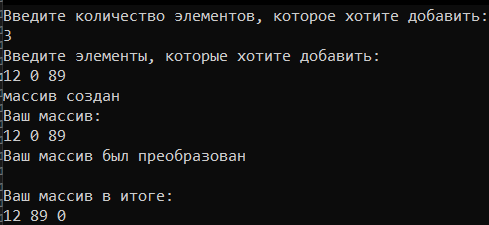
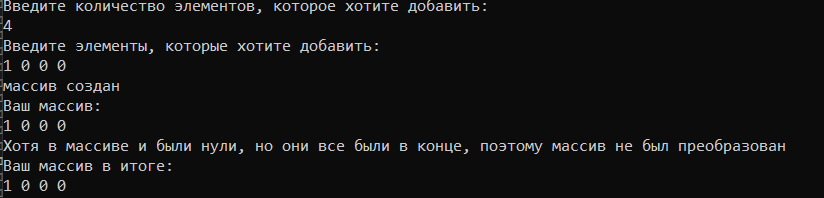
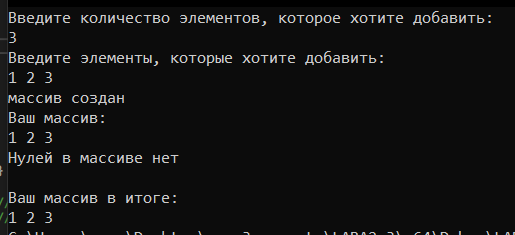
3.1 Описание разработанной программы

Данная программа позволяет пользователю создать собственную очередь, содержащую элементы типа char, добавить элементы в стек/очередь, очистить стек, просмотреть стек/очередь, преобразовать стек в очередь и завершить работу программы в режиме диалога.

3.2 Тестирование программы

Таблица 2 – Тесты программы

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тестируемая функция | Входные данные | Ожидаемый результат | Цель теста |
| 1 | create\_structure | N = a | некорректный ввод  Введите количество элементов, которое хотите добавить: | Некорректное число n |
| 2 | create\_structure | N=0 | Некорректное число n [1;10]  Введите количество элементов, которое хотите добавить: | Некорректное число n |
| 3 | create\_structure | N = 11 | Некорректное число n [1;10]  Введите количество элементов, которое хотите добавить: | Некорректное число n |
| 4 | create\_structure | N = 3  1 asd 67  Фы  Про  8 | Введите элементы, которые хотите добавить:  1 asd 67  некорректный ввод (запрос ввести заново)  фы  некорректный ввод  (запрос ввести заново)  про  некорректный ввод  (запрос ввести заново)  8  массив создан  Ваш массив:  1 67 8 | Некорректный элемент |
| 5 | MassivTransformation | N = 1  1 | Массив из одного элемента, преобразования не имеют смысл | Не смысла преобразовывать |
| 6 | MassivTransformation | n= 5  -12 21 54 45 43 | Нулей в массиве нет  Ваш массив в итоге:  -12 21 54 45 43 | Нет нулевых элементов, нет смысла преобразовывать |
| 7 | MassivTransformation | N = 3  12 0 89 | Ваш массив был преобразован  Ваш массив в итоге:  12 89 0 | Есть ноль, нужно переставить в конец массива |
| 8 | MassivTransformation | N = 4  1 0 0 0 | Хотя в массиве и были нули, но они все были в конце, поэтому массив не был преобразован  Ваш массив в итоге:  1 0 0 0 | Все нули уже в конце, нет смысла преобразовывать |
| 9 | MassivOutput | N = 3  1 2 3 | Ваш массив:  1 2 3 | Проверка вывода |

1. 
2. 
3. 
4. 
5. 
6. 
7. 
8. 
9. 

**Вывод:**

В данной лабораторной работе были реализованы

Также в данной лабораторной работе был обработан ряд исключений, таких как:

- Ошибка: превышен максимальный размер очереди!

- Очередь пуста!

- Ошибка: превышен максимальный размер стека!

- Ошибка! Необходимо ввести целое число, попробуйте еще раз

- Ошибка: введите только один символ.

- Ошибка: введите корректное положительное число

1. **Код**

#include <iostream>

#include <string>

#include <vector>

#include <windows.h> // SetConsoleOutputCP, SetConsoleCP

#include <stdio.h> //printf , fgets

#include <conio.h> // getch

#include <string.h> // strcpy, strncmp, strchr

#include <set>

using namespace std;

vector <int> add\_newelem(vector <int>&Massiv)

{

string newelem\_string;

cin.ignore(0, '\n') >> newelem\_string;

//cout << newelem<<endl;

try {

int newelem = stoi(newelem\_string);

//cout << newelem << endl;

try

{

Massiv.push\_back(newelem);

//cout << "элемент добавлен" << endl;

}

catch (...)

{

cout << "не удалось добавить элемент в структуру";

}

}

catch (...)

{

cout << "некорректный ввод" << endl;

Massiv = add\_newelem(Massiv);

}

return Massiv;

}

void create\_structure(vector <int>& Massiv)

{

if (!Massiv.empty()) cout << "изначальный массив не пустой, но мы добавим n элементов в него" << endl;

cout << "Введите количество элементов, которое хотите добавить: " << endl;

string n\_string;

cin.ignore(0, '\n') >> n\_string;

try {

int n = stoi(n\_string);

if (n < 1 or n >10)

{

cout << "Некорректное число n [1;10]" << endl;

cin.clear();

create\_structure(Massiv);

return;

}

cout << "Введите элементы, которые хотите добавить: " << endl;

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

{

Massiv = add\_newelem(Massiv);

}

cout << "массив создан" << endl;

}

catch (...)

{

cout << "некорректный ввод"<< endl;

create\_structure(Massiv);

return;

}

return;

}

void MassivOutput(vector <int> Massiv)

{

for (int i = 0; i < Massiv.size(); i++)

{

cout << Massiv[i] << " ";

}

return;

}

void MassivTransformation(vector <int>& Massiv, vector <int> InitialMassiv)

{

int count\_of\_null = 0;

for (int i = 0; i < Massiv.size(); i++)

{

if (Massiv[i] == 0)

{

count\_of\_null++;

//Massiv\_of\_null.push\_back(Massiv[i]);

Massiv.erase(Massiv.begin() + i);

i--;

}

}

if (count\_of\_null > 0)

{

for (int i = 0; i < count\_of\_null; i++) Massiv.push\_back(0);

if (Massiv == InitialMassiv) cout << endl << "Хотя в массиве и были нули, но они все были в конце, поэтому массив не был преобразован";

else cout << endl << "Ваш массив был преобразован" << endl;

}

else cout << endl << "Нулей в массиве нет" << endl;

return;

}

void main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

vector <int> Massiv;

create\_structure(Massiv);

vector <int> InitialMassiv;

InitialMassiv = Massiv;

cout << "Ваш массив:" << endl;

MassivOutput(Massiv);

if (Massiv.size() == 1) { cout << endl <<"Массив из одного элемента, преобразования не имеют смысл"; return; }

//vector <int> Massiv\_of\_null;

MassivTransformation(Massiv, InitialMassiv);

cout <<endl<< "Ваш массив в итоге: " << endl;

MassivOutput(Massiv);

//Massiv = Massiv + Massiv\_of\_null;

return;

}