

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования

**«Пермский национальный исследовательский
политехнический университет»**

Электротехнический факультет
Кафедра «Информационные технологии и автоматизированные системы»
направление подготовки: 09.03.01– «Информатика и вычислительная техника»

**Лабораторная работа № 10
по дисциплине
«Информатика»
на тему
«Динамические массивы»**

Выполнил студент гр. ИВТ-23-16

Южаков Федор Алексеевич

Проверил:

доцент кафедры ИТАС

Денис Владимирович Яруллин

(оценка)

(подпись)

(дата)

г. Пермь, 2024

ТЕОРЕТИЧЕСКАЯ ЧАСТЬ

1 Вариант задания

Постановка задачи.

Написать программу, в которой создаются динамические массивы и выполнить их обработку в соответствии со своим вариантом.

Порядок выполнения работы.

1. Ввести размер массива;
2. Сформировать массив с помощью операции new;
3. Заполнить массив (можно с помощью датчика случайных чисел);
4. Выполнить задание варианта, сформировать новый массив(ы)-результат(ы);
5. Напечатать массив(ы)-результат(ы);
6. Удалить динамические массивы с помощью операции delete.

Вариант 3.

Сформировать одномерный массив. Удалить из него K элементов, начиная с заданного индекса, добавить элемент с заданным ключом.

2 Анализ задачи

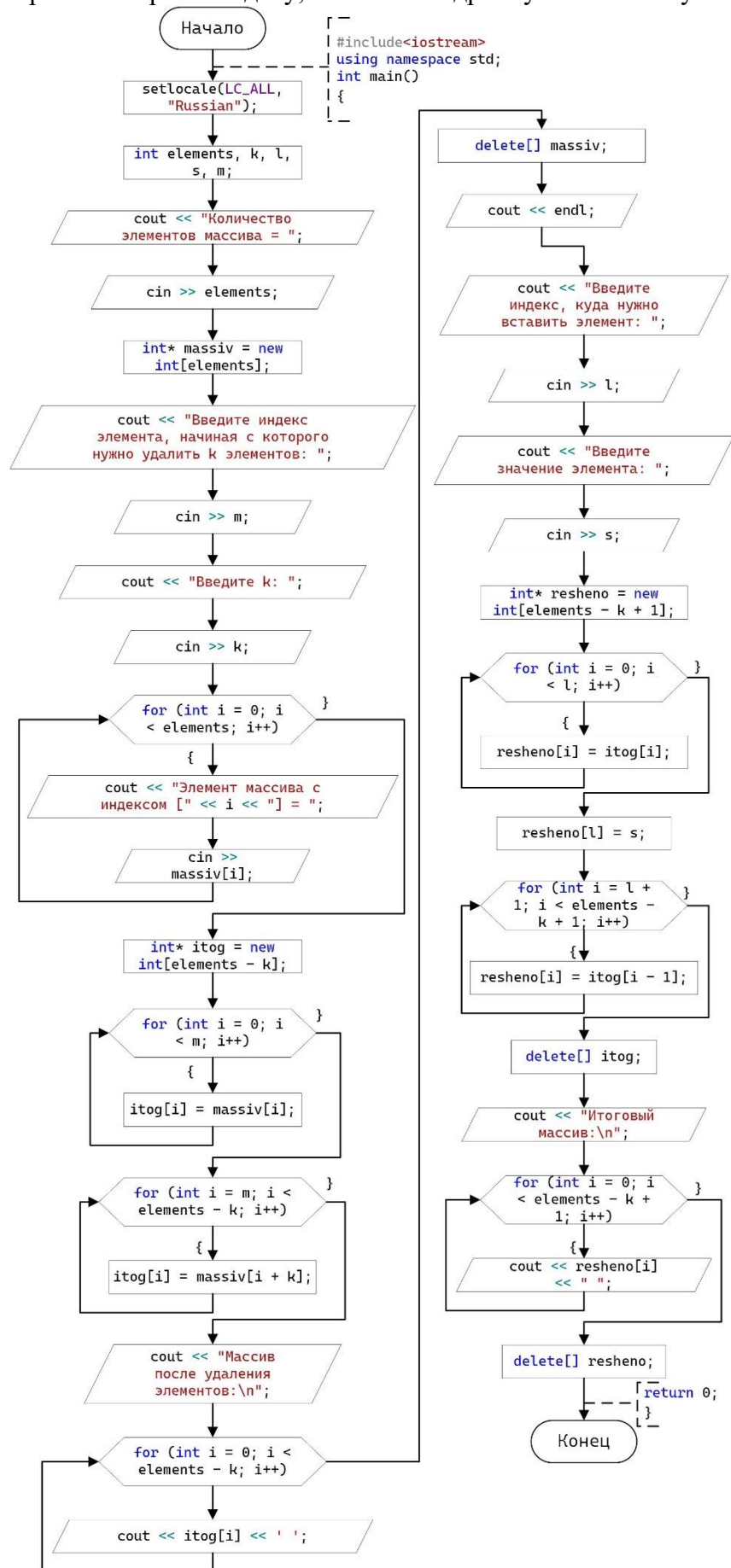
В данной задаче пользователь вводит количество элементов в массиве, индекс элемента, начиная с которого необходимо удалить K элементов, и само число K. Для удобства проверки пользователь также вводит все элементы изначального массива. После удаления элементов в массиве пользователю выводится новый массив. Пользователь вводит индекс элемента нового массива и его значение. Программа вставляет этот элемент на указанное место, а все остальные сдвигает вправо, увеличивая размер массива. После этого выводится окончательный массив. Все динамические массивы для экономии памяти удаляются по мере необходимости.

Удаление элементов происходит путём копирования всех элементов из изначального массива в новый, уменьшенный на K элементов, до элемента с указанным индексом. Далее, K элементов, считая элемент с указанным индексом, пропускаются, а оставшиеся копируются во второй массив. После этого первый массив можно удалить, а второй вывести.

Теперь пользователь вводит индекс и сам элемент, который нужно вставить. Программа создает третий массив, на один элемент больший, чем второй. В него копируются все элементы из второго, до указанного элемента. Потом вставляется введенный пользователем элемент, после него копируются оставшиеся элементы из второго массива. Второй массив удаляется, а третий выводится. После вывода удаляется и третий. Задача решена.

3 Блок схема

Проанализировав задачу, составим подробную блок схему основной программы.



ПРАКТИЧЕСКАЯ ЧАСТЬ

4 Результат решения

4.1 Готовая программа

Исходя из подробной блок-схемы, составим программу на языке C++.

Таблица 1 – Готовая программа задачи

```
#include<iostream>
using namespace std;
int main()
{
    setlocale(LC_ALL, "Russian");
    int elements, k, l, s, m;
    cout << "Количество элементов массива = ";
    cin >> elements;
    int* massiv = new int[elements];
    cout << "Введите индекс элемента, начиная с которого нужно
удалить k элементов: ";
    cin >> m;
    cout << "Введите k: ";
    cin >> k;
    for (int i = 0; i < elements; i++)
    {
        cout << "Элемент массива с индексом [" << i << "] = ";
        cin >> massiv[i];
    }
    int* itog = new int[elements - k];
    for (int i = 0; i < m; i++) itog[i] = massiv[i];
    for (int i = m; i < elements - k; i++) itog[i] = massiv[i +
k];
    cout << "Массив после удаления элементов:\n";
    for (int i = 0; i < elements - k; i++) cout << itog[i] << '
';
    delete[] massiv;
    cout << endl;
    cout << "Введите индекс, куда нужно вставить элемент: ";
    cin >> l;
    cout << "Введите значение элемента: ";
    cin >> s;
    int* resheno = new int[elements - k + 1];
    for (int i = 0; i < l; i++) resheno[i] = itog[i];
    resheno[l] = s;
    for (int i = l + 1; i < elements - k + 1; i++) resheno[i] =
itog[i - 1];
    delete[] itog;
    cout << "Итоговый массив:\n";
    for (int i = 0; i < elements - k + 1; i++) cout << resheno[i]
<< " ";
    delete[] resheno;
    return 0;
}
```

4.2 Скриншоты

Запустим программу, чтобы убедиться, что она работает верно.

```
Консоль отладки Microsoft V x + -
Количество элементов массива = 13
Введите индекс элемента, начиная с которого нужно удалить k элементов: 4
Введите k: 5
Элемент массива с индексом [0] = 23
Элемент массива с индексом [1] = 76
Элемент массива с индексом [2] = 24
Элемент массива с индексом [3] = 99
Элемент массива с индексом [4] = 123
Элемент массива с индексом [5] = 746
Элемент массива с индексом [6] = 235
Элемент массива с индексом [7] = 765
Элемент массива с индексом [8] = 888
Элемент массива с индексом [9] = 345
Элемент массива с индексом [10] = 2385
Элемент массива с индексом [11] = 23
Элемент массива с индексом [12] = 77
Массив после удаления элементов:
23 76 24 99 345 2385 23 77
Введите индекс, куда нужно вставить элемент: 4
Введите значение элемента: 0
Итоговый массив:
23 76 24 99 0 345 2385 23 77
C:\Users\Федор\Desktop\Б\Информатика\Лабы\10\10\x64\Debug\10.exe (процесс 13
16) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параме
тр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при
остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:|

Консоль отладки Microsoft V x + -
Количество элементов массива = 14
Введите индекс элемента, начиная с которого нужно удалить k элементов: 5
Введите k: 7
Элемент массива с индексом [0] = 78
Элемент массива с индексом [1] = -45
Элемент массива с индексом [2] = 6666
Элемент массива с индексом [3] = 0
Элемент массива с индексом [4] = 54
Элемент массива с индексом [5] = -12
Элемент массива с индексом [6] = 65
Элемент массива с индексом [7] = 77
Элемент массива с индексом [8] = 77
Элемент массива с индексом [9] = 876
Элемент массива с индексом [10] = -34
Элемент массива с индексом [11] = 127
Элемент массива с индексом [12] = 777
Элемент массива с индексом [13] = -777
Массив после удаления элементов:
78 -45 6666 0 54 777 -777
Введите индекс, куда нужно вставить элемент: 7
Введите значение элемента: 5
Итоговый массив:
78 -45 6666 0 54 777 -777 5
C:\Users\Федор\Desktop\Б\Информатика\Лабы\10\10\x64\Debug\10.exe (процесс 18492)
завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "
Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль при остановк
е отладки".
Нажмите любую клавишу, чтобы закрыть это окно:|

Консоль отладки Microsoft V x + -
Количество элементов массива = 3
Введите индекс элемента, начиная с которого нужно удалить k элементов: 1
Введите k: 0
Элемент массива с индексом [0] = 234
Элемент массива с индексом [1] = 654
Элемент массива с индексом [2] = 333
Массив после удаления элементов:
234 654 333
Введите индекс, куда нужно вставить элемент: 0
Введите значение элемента: 444
Итоговый массив:
444 234 654 333
C:\Users\Федор\Desktop\Б\Информатика\Лабы\10\10\x64\Debug\10.exe (процесс 1
9676) завершил работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите парам
етр "Сервис" ->"Параметры" ->"Отладка" -> "Автоматически закрыть консоль пр
и остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно:|
```

ВЫВОД

В итоге этой работы была составлена программа с использованием динамических массивов. Сделан вывод о полезности и сферах применения таких массивов. Изучены тонкости работы с динамическими массивами. Выявлены плюсы и минусы динамических массивов перед статическими. В ходе работы были получены навыки работы с динамическими массивами.

Проведенная лабораторная работа была опубликована в общий доступ по адресу: https://github.com/Fedor0000/TheUltimateFolder/tree/main/Sem_2/Labs/10_3ЕЛ