Пермский национальный исследовательский политехнический университет

Кафедра информационные технологии и автоматизированные системы

Лабораторная работа №10

“**Динамические массивы**”

|  |  |  |
| --- | --- | --- |
|  |  | Выполнил:  студент группы: ИВТ-24-2б  К.Д. Горбушин  Проверил:  доцент кафедры ИТАС  О.А. Полякова |

Пермь 2025

# Постановка задачи

Сформировать одномерный массив. Удалить из него элемент с заданным номером, добавить К элементов, начиная с заданного номера;

**Код программы**

#include <iostream>

using namespace std;

// Функция для удаления элемента из массива по индексу

int\* deleteElement(int\* arr, int& size, int indexToDelete) {

if (indexToDelete < 0 || indexToDelete >= size) {

cout << "Ошибка: Неверный индекс для удаления." << endl;

return arr; // Возвращаем исходный массив, если индекс некорректен

}

int\* newArr = new int[size - 1]; // Создаем новый массив на 1 элемент меньше

for (int i = 0, j = 0; i < size; ++i) {

if (i != indexToDelete) {

newArr[j++] = arr[i];

}

}

delete[] arr; // Освобождаем память, выделенную под старый массив

size--; // Уменьшаем размер

return newArr;

}

// Функция для добавления K элементов, начиная с заданного номера

int\* addElements(int\* arr, int& size, int indexToAdd, int\* elementsToAdd, int k) {

if (indexToAdd < 0 || indexToAdd > size) {

cout << "Ошибка: Неверный индекс для вставки." << endl;

return arr; // Возвращаем исходный массив, если индекс некорректен

}

int\* newArr = new int[size + k]; // Создаем новый массив на K элементов больше

// Копируем элементы до индекса вставки

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

newArr[i] = arr[i];

}

// Вставляем новые элементы

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

newArr[indexToAdd + i] = elementsToAdd[i];

}

// Копируем оставшиеся элементы после вставки

for (int i = indexToAdd; i < size; ++i) {

newArr[indexToAdd + k + (i - indexToAdd)] = arr[i];

}

delete[] arr; // Освобождаем память, выделенную под старый массив

size += k; // Увеличиваем размер

return newArr;

}

int main() {

int size;

setlocale(LC\_ALL, "Ru");

cout << "Введите размер массива: ";

cin >> size;

int\* arr = new int[size]; // Создаем динамический массив

cout << "Введите элементы массива:" << endl;

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

cin >> arr[i];

}

// Удаление элемента

int indexToDelete;

cout << "Введите индекс элемента для удаления: ";

cin >> indexToDelete;

arr = deleteElement(arr, size, indexToDelete);

// Вывод массива после удаления

cout << "Массив после удаления элемента: ";

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

cout << arr[i] << " ";

}

cout << endl;

// Добавление элементов

int indexToAdd, k;

cout << "Введите индекс для вставки новых элементов: ";

cin >> indexToAdd;

cout << "Введите количество элементов для добавления: ";

cin >> k;

int\* elementsToAdd = new int[k]; // Создаем массив для новых элементов

cout << "Введите элементы для добавления:" << endl;

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

cin >> elementsToAdd[i];

}

arr = addElements(arr, size, indexToAdd, elementsToAdd, k);

delete[] elementsToAdd; // Освобождаем память после использования

// Вывод массива после добавления

cout << "Массив после добавления элементов: ";

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

cout << arr[i] << " ";

}

cout << endl;

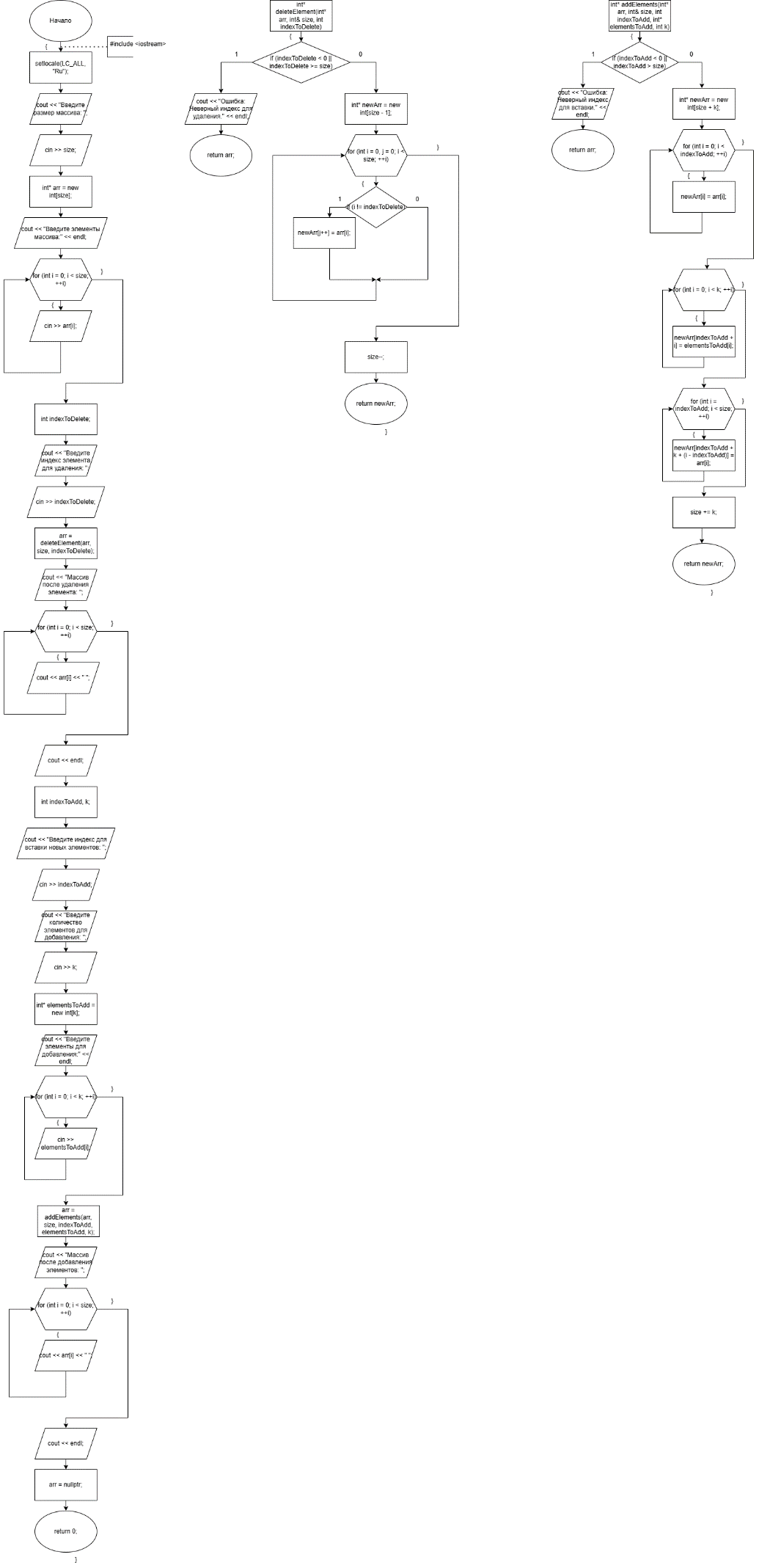
delete[] arr; // Освобождаем память, выделенную под массив

arr = nullptr;

return 0;

}

**Блок-схема**

****

**Ссылка на GITHUB:** [https://github.com/kirill2209](https://github.com/kirill2209%20)