

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«Пермский национальный исследовательский политехнический университет»
(ПНИПУ)
Электротехнический факультет
Кафедра «Информационные технологии и автоматизированные системы»
(ИТАС)

Лабораторная работа
на тему
«Динамические массивы»

Выполнил
Студент группы ИВТ-23-16
Адаев Даниил Дмитриевич
Проверил
Доцент кафедры ИТАС
Д. В. Яруллин

Постановка задачи

Вариант 2

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` или библиотечных функций `malloc` (`calloc`);
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` или библиотечной функции `free`.

Сформировать одномерный массив. Удалить из него элемент с заданным ключом, добавить элемент с заданным ключом;

Словесный алгоритм

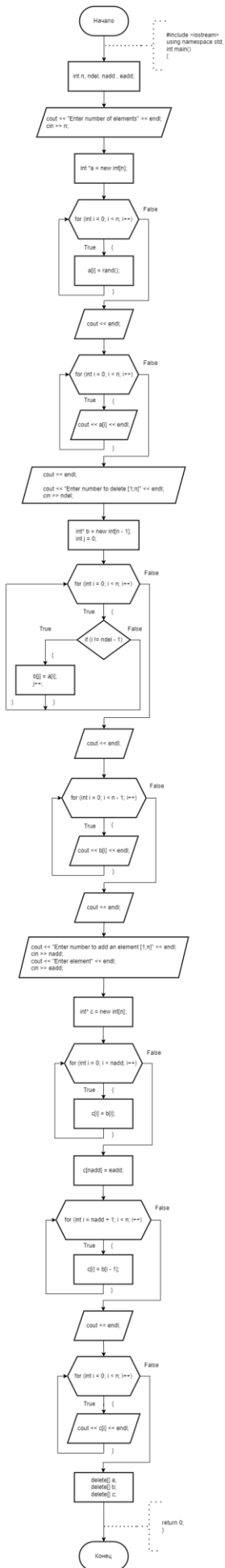
Целочисленные переменные: **n** – размер массива, **ndelm** – номер удаляемого элемента, **nadd** – номер добавляемого элемента, **eadd** – значение удаляемого элемента.

Алгоритм запрашивает количество элементов, затем заполняет динамический массив **a** случайными числами и выводит его.

Алгоритм запрашивает номер удаляемого элемента, создает динамический массив **b** и заполняет его значениями массива **a**, кроме элемента с введенным номером. Массив **b** выводится.

Алгоритм запрашивает номер добавляемого элемента и его значение, создает динамический массив **c** и до полученного номера заполняет значениями массива **b**, вводит значение добавляемого элемента и заполняет оставшиеся значения массива **b**.

Освобождается память всех динамических массивов.



Код программы

```
#include <iostream>

using namespace std;

int main()
{
    int n, ndel, nadd, eadd;

    cout << "Enter number of elements" << endl;
    cin >> n;

    int *a = new int[n];

    for (int i = 0; i < n; i++)
    {
        a[i] = rand();
    }

    cout << endl;

    for (int i = 0; i < n; i++)
    {
        cout << a[i] << endl;
    }

    cout << endl;

    cout << "Enter number to delete [1;n]" << endl;
    cin >> ndel;

    int* b = new int[n - 1];
    int j = 0;

    for (int i = 0; i < n; i++)
    {
        if (i != ndel - 1)
        {
            b[j] = a[i];
            j++;
        }
    }

    cout << endl;

    for (int i = 0; i < n - 1; i++)
    {
        cout << b[i] << endl;
    }

    cout << endl;

    cout << "Enter number to add an element [1;n]" << endl;
    cin >> nadd;
    cout << "Enter element" << endl;
    cin >> eadd;

    int* c = new int[n];

    for (int i = 0; i < nadd; i++)
    {
        c[i] = b[i];
    }
```

```

        c[nadd] = eadd;

    for (int i = nadd + 1; i < n; i++)
    {
        c[i] = b[i - 1];
    }

    cout << endl;

    for (int i = 0; i < n; i++)
    {
        cout << c[i] << endl;
    }

    delete[] a;
    delete[] b;
    delete[] c;

    return 0;
}

```

Выводы

```

Enter number of elements
5

41
18467
6334
26500
19169

Enter number to delete [1;n]
3

41
18467
26500
19169

Enter number to add an element [1;n]
2
Enter element
-1

41
18467
-1
26500
19169

```