

## Домашнее задание “Динамический массив” 2

В дополнение к функциям из задания “Динамический массив 1” разработать функции:

- Вставки элемента на указанную позицию
- Удаления элемента из указанной позиции
- Сортировки массива
- Поиска элемента в массиве по его значению

Функции добавления элемента в конец массива и удаления последнего элемента массива переделать под использование ссылок.

Функции с одномерными динамическим массивом из первого задания:

- Создания динамического массива заданного размера .
- Заполнения (инициализации) динамического массива случайными значениями .
- Печати динамического массива.
- Удаления динамического массива.
- Добавления элемента в конец массива.
- Удаления последнего элемента из массива и возврат его значения.

Прототипы функций и пример работы с динамическим массивом:

```
#include <iostream>

using namespace std;

int* createArray(int arraySize);
void fillArrayRandom(int * arr, int arraySize, int min, int max);
void printArray(const int * arr, int arraySize);
void deleteArray(int * arr);
void pushItemBack(int ** arr, int& arraySize, int newItem);
int popItem(int ** arr, int& arraySize); // return popped item

void InsertItemAt(int ** arr, int& arraySize, int newItem, int position);
void RemoveItemAt(int ** arr, int& arraySize, int position);
void SortArray(int * arr, int arraySize);
int FindItem(const int * arr, int arraySize, int valueToFind); // return position or -1

int main() {
    int arraySize;
    cin >> arraySize;

    int *dynArr = createArray(arraySize);
    fillArrayRandom(dynArr, arraySize, 0, 100);
    printArray(dynArr, arraySize);

    for(int i = 0, sz = arraySize; i < sz; i++ )
        pushItemBack(&dynArr, arraySize, dynArr[i]);
    printArray(dynArr, arraySize);

    for(int i = 0, sz = arraySize/2; i < sz; i++ )
        cout << popItem(&dynArr, arraySize);

    deleteArray(dynArray);
}
```