



Министерство науки и высшего образования Российской Федерации
Мытищинский филиал
Федерального государственного бюджетного образовательного учреждения
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ космический

КАФЕДРА К-3

ОТЧЕТ
К ЛАБОРАТОРНОЙ РАБОТЕ

№ 3

по ДИСЦИПЛИНЕ
«ИНФОРМАТИКА»

Студент К3-13Б
(Группа)

И.И. Остапенко
(И.О.Фамилия)

Преподаватель

В.В. Афанасьева
(И.О.Фамилия)

2022 г.

Задание № 3 (лабораторная работа №3)

Задача: дан массив вещественных чисел. С клавиатуры вводится число X, определить находится ли это число X в заданном массиве (если находится, то указать его место в массиве) с помощью поиска «перебором».

Необходимо:

- 1) Определить входные и выходные данные.
- 2) Разработать блок-схему алгоритма.
- 3) Реализовать алгоритм на языке C.
- 4) Составить тесты для проверки работоспособности алгоритма.
- 5) Проверить работоспособность программы на разработанных тестах.
- 6) Оформить отчет по лабораторной работе (см. требования к отчету).

Указание: массив вводить с клавиатуры, предварительно ввести с клавиатуры количество элементов массива (делать проверку, чтобы количество элементов массива было больше или равно 1), желательно память под массив выделять динамически. Массив можно задавать как упорядоченным, так и неупорядоченным.

ТЗ:

Входные данные:

n – целое число, длина массива;

ar[] – массив вещественных чисел;

x – вещественное число, номер которого надо найти;

Выходные данные: index – целое число, номер искомого x в массиве;

Функция makeArray возвращает вещественный массив вещественных чисел:

n – целое число, длина массива;

ar[] – динамический массив вещественных чисел;

i – текущий номер элемента массива, целое число;

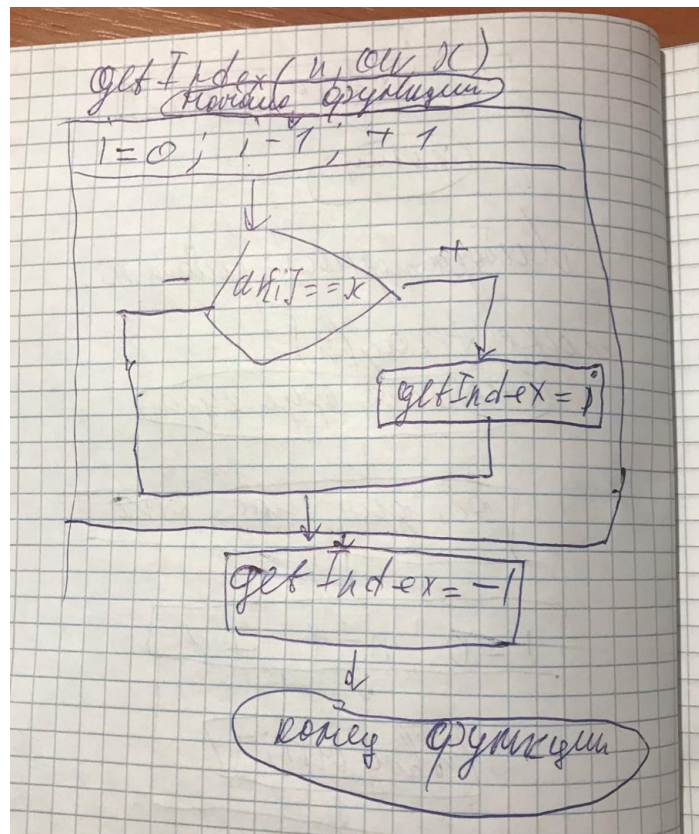
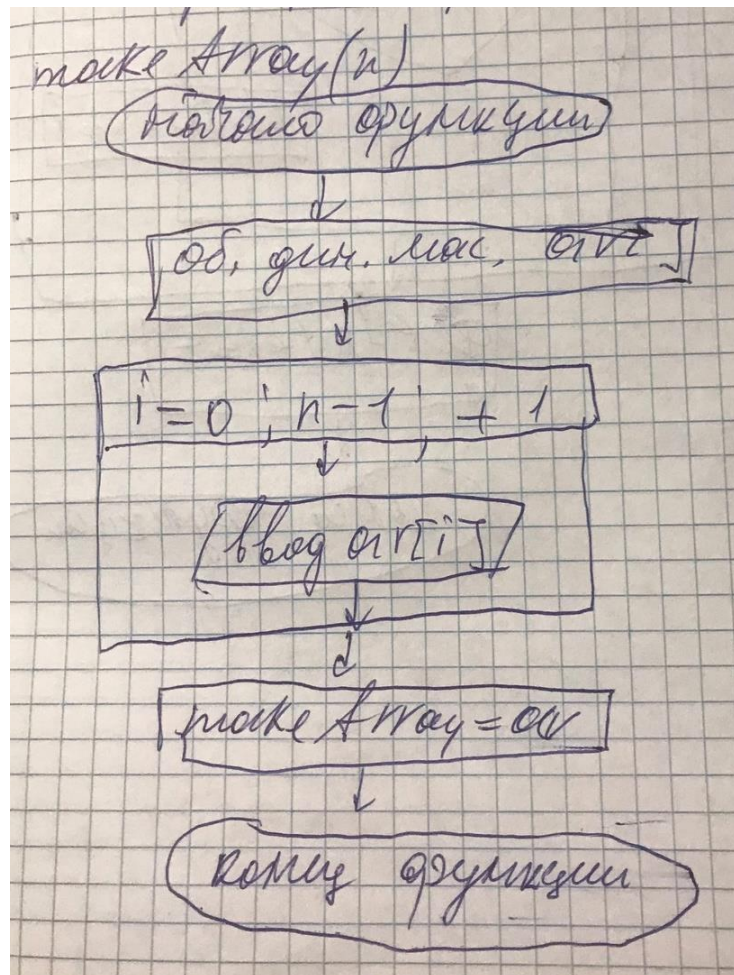
Функция getIndex возвращает целое число:

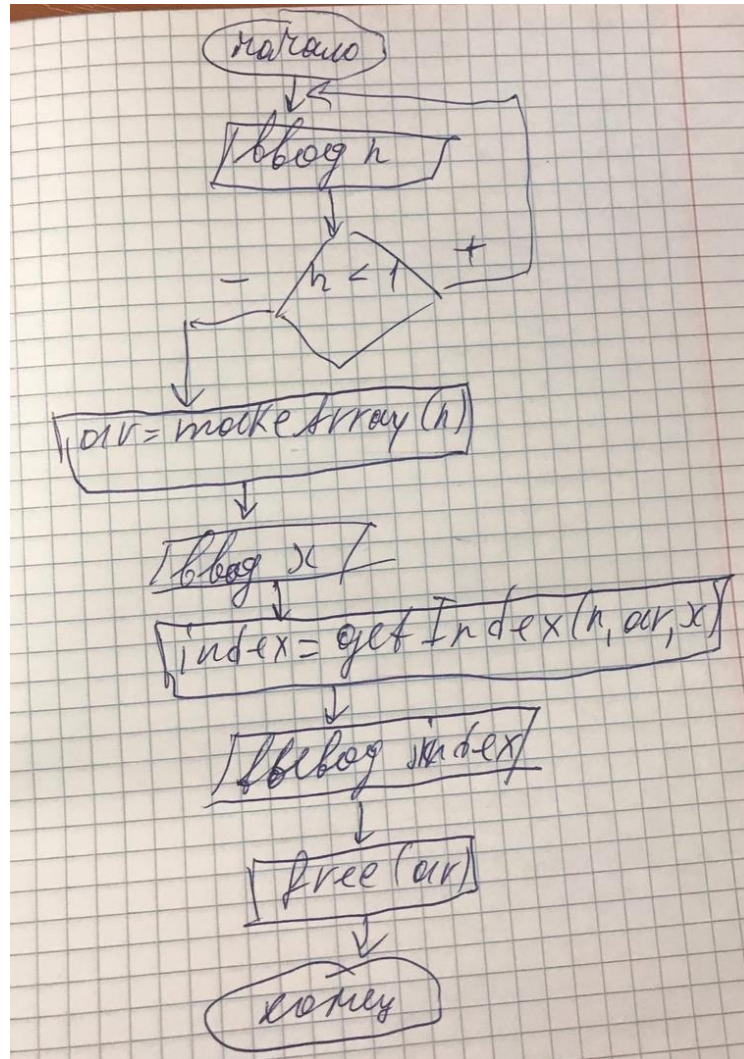
n – целое число, длина массива;

ar[] – массив вещественного типа;

x – искомое число;

i – текущий номер элемента массива, целое число;





```
#include <stdio.h>
#include <stdlib.h>
```

```
double* makeArray(int n) {
    double* ar = (double*)malloc(sizeof(double) * n);
    for (int i = 0; i < n; i++) {
        printf("ar[%d]: ", i);
        scanf("%lf", &ar[i]);
    }
    return ar;
}
```

```
int getIndex(int n, double* ar, double x) {
    for (int i = 0; i < n; i++) {
        if (ar[i] == x) {
            return i;
        }
    }
    return -1;
}
```

```

int main()
{
    int n;
    do {
        printf("Enter n: ");
        scanf("%d", &n);
    } while (n < 1);
    double* ar = makeArray(n);
    double x;
    printf("Enter x: ");
    scanf("%lf", &x);
    int index = getIndex(n, ar, x);
    printf("%d", index);
    free(ar);
    return 0;
}

```

```

Enter n: 0
Enter n:

```

```

Enter n: 1
ar[0]: 1
Enter x: 1
0

```

```

Enter n: 3
ar[0]: 1
ar[1]: 2
ar[2]: 3
Enter x: 3
2

```

```

Enter n: 4
ar[0]: 1
ar[1]: 324
ar[2]: -99.45454
ar[3]: 23
Enter x: -99.4
-1

```

```

Enter n: 5
ar[0]: 23
ar[1]: 34
ar[2]: 2
ar[3]: -556.5665
ar[4]: 4
Enter x: 2
2

```