



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

№ 2

по ДИСЦИПЛИНЕ
«ИНФОРМАТИКА»

Студент К3-13Б
(Группа)

И.И. Остапенко
(И.О.Фамилия)

Преподаватель

В.В. Афанасьева
(И.О.Фамилия)

2022 г.

Задание № 2 (лабораторная работа №2)

Задача: дан массив вещественных чисел, поменять в этом массиве максимальный и минимальный элементы местами.

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.

ТЗ:

Входные данные: n-целое число, длина массива; ar[]-динамический массив вещественных чисел;

Выходные данные: ar[]-изменённый массив;

imax – номер максимального элемента, целое число;

imin – номер минимального элемента, целое число;

функция makeArray возвращает массив вещественных чисел:

n – целое число, длина массива;

ar[] – динамический массив вещественных чисел;

i – текущий номер элемента массива

функция getIMax возвращает целое число:

n – целое число, длина массива;

ar[] – динамический массив вещественных чисел;

max – вещественное число, хранит максимальный элемент массива;

imax – целое число, хранит номер максимального элемента массива;

функция getIMin возвращает целое число:

n – целое число, длина массива;

ar[] – динамический массив вещественных чисел;

min – вещественное число, хранит минимальный элемент массива;

imin – целое число, хранит номер минимального элемента массива;

функция changeMaxMin возвращает массив вещественных чисел:

ar[] – динамический массив вещественных чисел;

imax – целое число, номер максимального элемента массива;

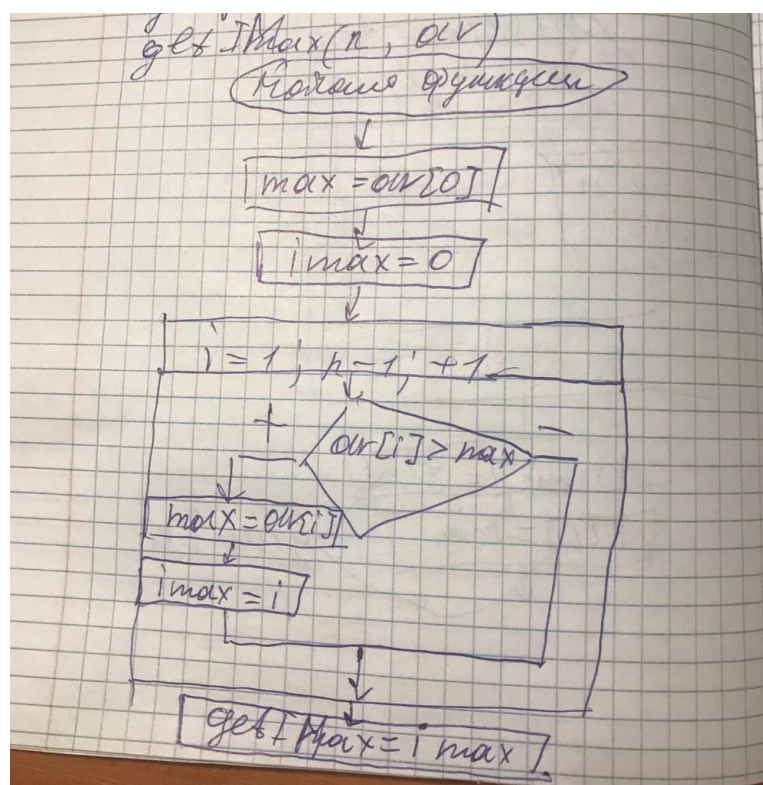
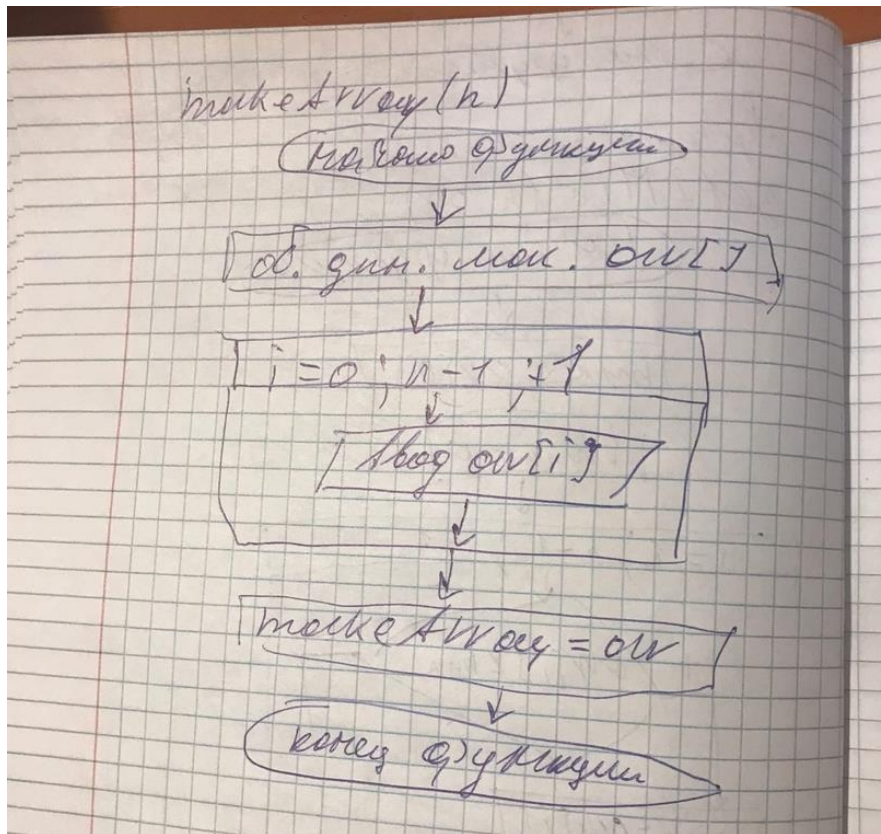
imin – целое число, номер минимального элемента массива;

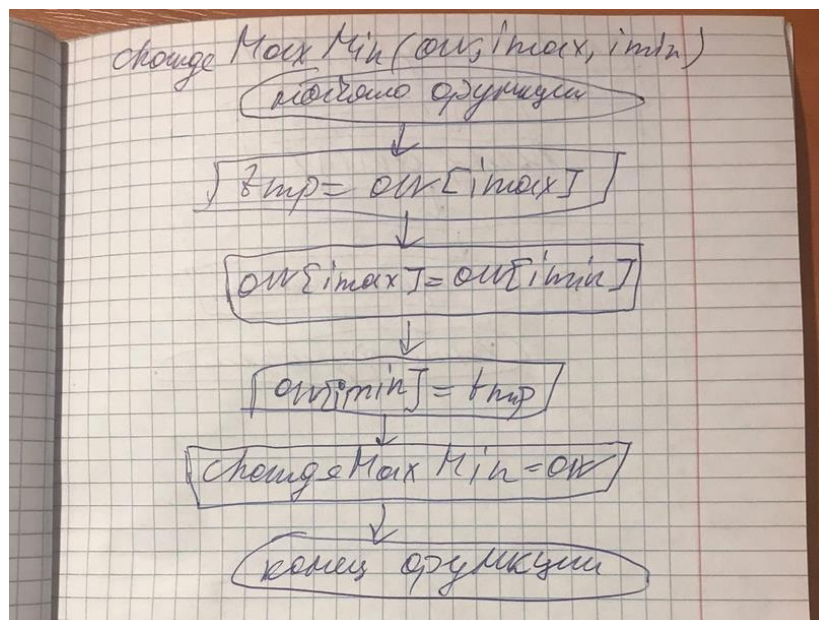
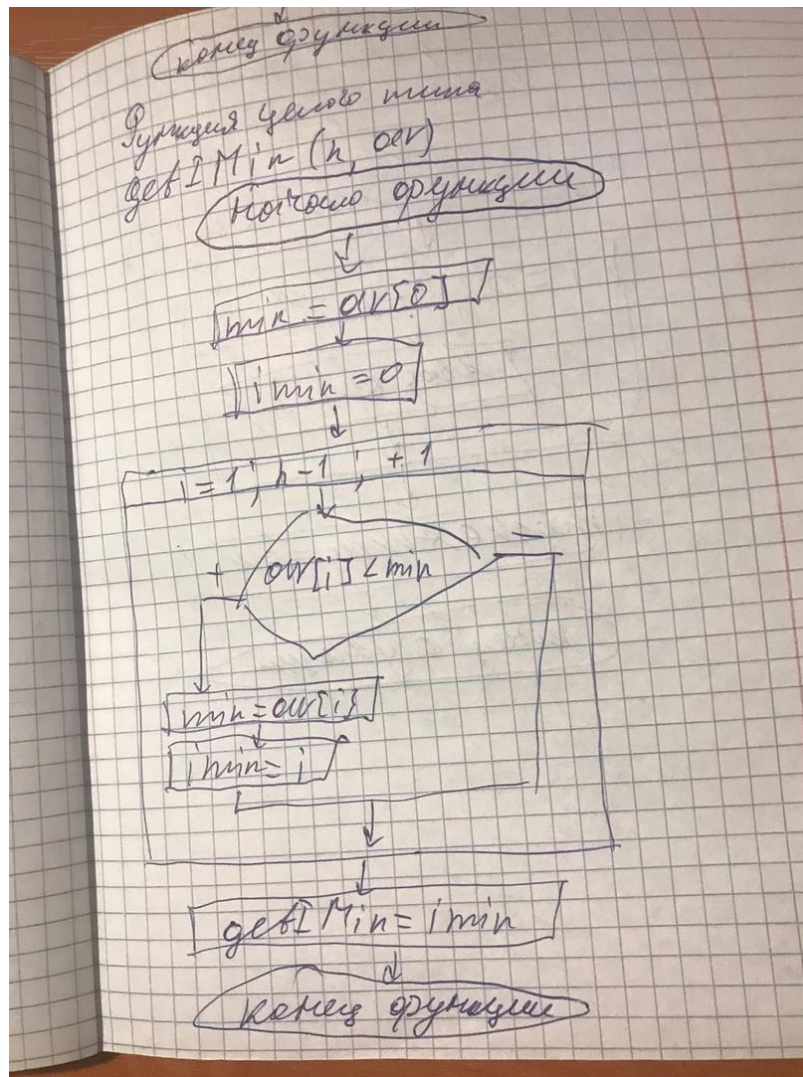
tmp – вещественное число, служит для обмена значениями элементов с индексами imin и imax

процедура printArray:

n – целое число, длина массива;

ar[] – динамический массив вещественных чисел;






```

#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("Enter ar[%d]: ", i);
        scanf("%lf", &ar[i]);
    }
    return ar;
}

int getIMax(int n, double* ar) {
    double max = ar[0];
    int imax = 0;
    for (int i = 1; i < n; i++) {
        if (ar[i] > max) {
            max = ar[i];
            imax = i;
        }
    }
    return imax;
}

int getIMin(int n, double* ar) {
    double min = ar[0];
    int imin = 0;
    for (int i = 1; i < n; i++) {
        if (ar[i] < min) {
            min = ar[i];
            imin = i;
        }
    }
    return imin;
}

double* changeMaxMin(double* ar, int imax, int imin) {
    double tmp = ar[imax];
    ar[imax] = ar[imin];
    ar[imin] = tmp;
    return ar;
}

void printArray(int n, double* ar) {
    for (int i = 0; i < n; i++) {
        printf("%f ", ar[i]);
    }
    free(ar);
}

```

```

int main()
{
    int n;
    do {
        printf("Enter n: ");
        scanf("%d", &n);
    } while (n < 1);
    double* ar = makeArray(n);
    int imax = getIMax(n, ar);
    int imin = getIMin(n, ar);
    ar = changeMaxMin(ar, imax, imin);
    printArray(n, ar);
    return 0;
}

```

```

Enter n: -1
Enter n:

```

```

Enter n: 1
Enter ar[0]: 1
1.000000

```

```

Enter n: 5
Enter ar[0]: 6.253434
Enter ar[1]: 7.878787878
Enter ar[2]: 3.45454543
Enter ar[3]: 2.32323333333
Enter ar[4]: 3.5
6.253434 2.323233 3.454545 7.878788 3.500000

```

```

Enter n: 4
Enter ar[0]: -10
Enter ar[1]: -123.4545454
Enter ar[2]: 67.68
Enter ar[3]: 159.6564566546
-10.000000 159.656457 67.680000 -123.454545

```