

## Лабораторная работа №4

**Тема:** Разработка и отладка программ обработки массивов.

**Цель занятия:** Выработать умения и навыки работы со средой программирования Microsoft Visual Studio.

**Время выполнения:** 2 часа.

### *Содержание работы и последовательность ее исполнения.*

1. Запустить Microsoft Visual Studio выбрав пункт меню «Пуск → Все программы → Microsoft Visual Studio → Microsoft Visual Studio».
2. Для каждого задания создавать отдельный пустой консольный проект (см. л.р. 1).

### **Задание 1. Одномерные массивы**

#### **Вариант 1**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- сумму отрицательных элементов массива;
- произведение элементов массива, расположенных между максимальным и минимальным элементами.

Упорядочить элементы массива по возрастанию.

#### **Вариант 2**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- сумму положительных элементов массива;
- произведение элементов массива, расположенных между максимальным по модулю и минимальным по модулю элементами.

Упорядочить элементы массива по убыванию.

#### **Вариант 3**

В одномерном массиве, состоящем из  $n$  целочисленных элементов, вычислить:

- произведение элементов массива с четными номерами;
- сумму элементов массива, расположенных между первым и последним нулевыми элементами.

Преобразовать массив таким образом, чтобы сначала располагались все положительные элементы, а потом – все отрицательные (элементы, равные нулю, считать положительными).

#### **Вариант 4**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- сумму элементов массива с нечетными номерами;
- сумму элементов массива, расположенных между первым и последним отрицательными элементами.

Сжать массив, удалив из него все элементы, модуль которых не превышает единицу. Освободившиеся в конце массива элементы заполнить нулями.

#### **Вариант 5**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- максимальный элемент массива;
- сумму элементов массива, расположенных до последнего положительного элемента.

Сжать массив, удалив из него все элементы, модуль которых находится в интервале  $[a, b]$ . Освободившиеся в конце массива элементы заполнить нулями.

#### **Вариант 6**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- минимальный элемент массива;
- сумму элементов массива, расположенных между первым и последним положительными элементами.

Преобразовать массив таким образом, чтобы сначала располагались все элементы, равные нулю, а потом – все остальные.

**Вариант 7**

В одномерном массиве, состоящем из  $n$  целочисленных элементов, вычислить:

- номер максимального элемента массива;
- произведение элементов массива, расположенных между первым и вторым нулевыми элементами.

Преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в нечетных позициях, а во второй половине – элементы, стоявшие в четных позициях.

**Вариант 8**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- номер минимального элемента массива;
- сумму элементов массива, расположенных между первым и вторым отрицательными элементами.

Преобразовать массив таким образом, чтобы сначала располагались все элементы, модуль которых не превышает единицу, а потом – все остальные.

**Вариант 9**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- максимальный по модулю элемент массива;
- сумму элементов массива, расположенных между первым и вторым положительными элементами.

Преобразовать массив таким образом, чтобы элементы, равные нулю, располагались после всех остальных.

**Вариант 10**

В одномерном массиве, состоящем из  $n$  целочисленных элементов, вычислить:

- минимальный по модулю элемент массива;
- сумму модулей элементов массива, расположенных после первого элемента, равного нулю.

Преобразовать массив таким образом, чтобы в первой его половине располагались элементы, стоявшие в четных позициях, а во второй половине – элементы, стоявшие в нечетных позициях.

**Вариант 11**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- номер минимального по модулю элемента массива;
- сумму модулей элементов массива, расположенных после первого отрицательного элемента.

Сжать массив, удалив из него все элементы, величина которых находится в интервале  $[a, b]$ . Освободившиеся в конце массива элементы заполнить нулями.

**Вариант 12**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- номер максимального по модулю элемента массива;
- сумму элементов массива, расположенных после первого положительного элемента.

Преобразовать массив таким образом, чтобы сначала располагались все элементы, целая часть которых лежит в интервале  $[a, b]$ , а потом – все остальные.

**Вариант 13**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- количество элементов массива, лежащих в диапазоне от  $A$  до  $B$ ;
- сумму элементов массива, расположенных после максимального элемента.

Упорядочить элементы массива по убыванию модулей.

**Вариант 14**

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- количество элементов массива, равных нулю;
- сумму элементов массива, расположенных после минимального элемента.

Упорядочить элементы массива по возрастанию модулей.

## Вариант 15

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- количество элементов массива, больших  $C$ ;
- произведение элементов массива, расположенных после максимального по модулю элемента.

Преобразовать массив таким образом, чтобы сначала располагались все отрицательные элементы, а потом – все положительные (элементы, равные нулю, считать положительными).

### Пример выполнения:

В одномерном массиве, состоящем из  $n$  вещественных элементов, вычислить:

- количество отрицательных элементов массива;
- сумму модулей элементов массива, расположенных после минимального по модулю элемента.

Заменить все отрицательные элементы массива их квадратами и упорядочить элементы массива по возрастанию.

```
#include <stdio.h> // для printf() и scanf()
#include <stdlib.h> // для rand()
#include <math.h>   // для abs()
int main() {
    int n;
    int m[100]; // массив из 100 элементов
    // вводим размер массива
    printf("N="); scanf("%i",&n);

    // заполнение и вывод массива
    printf("Начальный массив:\n");
    for(int i=0; i<n; i++) {
        m[i]=rand()%50-25;
        printf("%4i",m[i]);
    }
    printf("\n");

    // подсчитываем кол-во отрицательных чисел
    int cnt=0;
    for(int i=0; i<n; i++) if(m[i]<0) cnt++;
    printf("Количество отрицательных чисел: %i\n",cnt);

    // поиск последнего минимального по модулю элемента
    int min=0;
    for(int i=1; i<n; i++) if(abs(m[min])>abs(m[i])) min=i;
    printf("Позиция последнего минимального числа: %i\n",min+1);

    // подсчитываем сумму модулей после минимального
    int summ=0;
    for(int i=min+1; i<n; i++) summ+=abs(m[i]);
    printf("Сумма модулей чисел после минимального: %i\n",summ);

    // заменяем отрицательные числа квадратами
    printf("Массив после коррекции отрицательных чисел:\n");
    for(int i=0; i<n; i++) {
        if(m[i]<0) m[i]*=m[i];
        printf("%4i",m[i]);
    }
    printf("\n");

    // сортируем массив
    for(int i=0; i<n; i++)
        for(int j=0; j<n; j++)
            if(m[i]<m[j]) {int tmp=m[i]; m[i]=m[j]; m[j]=tmp;}
    printf("Сортированный массив:\n");
    for(int i=0; i<n; i++)
        printf("%4i",m[i]);
    printf("\n");

    return 0;
}
```

## Задание 2. Двумерные массивы

### Вариант 1

Дана целочисленная прямоугольная матрица. Определить:

- количество строк, не содержащих ни одного нулевого элемента;
- максимальное из чисел, встречающихся в заданной матрице более одного раза.

### Вариант 2

Дана целочисленная прямоугольная матрица. Определить количество столбцов, не содержащих ни одного нулевого элемента. Характеристикой строки целочисленной матрицы назовем сумму ее положительных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с ростом характеристик.

### Вариант 3

Дана целочисленная прямоугольная матрица. Определить:

- количество столбцов, содержащих хотя бы один нулевой элемент;
- номер строки, в которой находится самая длинная серия одинаковых элементов.

### Вариант 4

Дана целочисленная квадратная матрица. Определить:

- произведение элементов в тех строках, которые не содержат отрицательных элементов;
- максимум среди сумм элементов диагоналей, параллельных главной диагонали матрицы.

### Вариант 5

Дана целочисленная квадратная матрица. Определить:

- сумму элементов в тех столбцах, которые не содержат отрицательных элементов;
- минимум среди сумм модулей элементов диагоналей, параллельных побочной диагонали матрицы.

### Вариант 6

Дана целочисленная прямоугольная матрица. Определить:

- сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент;
- номера строк и столбцов всех седловых точек матрицы.

**Примечание:** Матрица  $A$  имеет седловую точку  $A_{ij}$ , если  $A_{ij}$  является минимальным элементом в  $i$ -й строке и максимальным – в  $j$ -м столбце.

### Вариант 7

Для заданной матрицы размером  $8 \times 8$  найти такие  $k$ , при которых  $k$ -я строка матрицы совпадает с  $k$ -м столбцом. Найти сумму элементов в тех строках, которые содержат хотя бы один отрицательный элемент.

### Вариант 8

Характеристикой столбца целочисленной матрицы назовем сумму модулей его отрицательных нечетных элементов. Переставляя столбцы заданной матрицы, расположить их в соответствии с ростом характеристик. Найти сумму элементов в тех столбцах, которые содержат хотя бы один отрицательный элемент.

### Вариант 9

Соседями элемента  $A_{ij}$  в матрице назовем элементы  $A_{kl}$ , где  $i-1 \leq k \leq i+1$ ,  $j-1 \leq l \leq j+1$ ,  $(k, l) \neq (i, j)$ . Операция сглаживания матрицы дает новую матрицу того же размера, каждый элемент которой получается как среднее арифметическое имеющихся соседей соответствующего элемента исходной матрицы. Построить результат сглаживания заданной вещественной матрицы размером  $10 \times 10$ . В сглаженной матрице найти сумму модулей элементов, расположенных ниже главной диагонали.

### Вариант 10

Элемент матрицы называется локальным минимумом, если он строго меньше всех имеющихся у него соседей (определение соседних элементов см. в варианте 9). Подсчитать количество локальных минимумов заданной матрицы размером  $10 \times 10$ . Найти сумму модулей элементов, расположенных выше главной диагонали.

**Вариант 11**

Коэффициенты системы линейных уравнений заданы в виде прямоугольной матрицы. С помощью допустимых преобразований привести систему к треугольному виду. Найти количество строк, среднее арифметическое элементов которых меньше заданной величины.

**Вариант 12**

Уплотнить заданную матрицу, удаляя из нее строки и столбцы, заполненные нулями. Найти номер первой из строк, содержащих хотя бы один положительный элемент.

**Вариант 13**

Осуществить циклический сдвиг элементов прямоугольной матрицы на  $n$  элементов вправо или вниз (в зависимости от введенного режима),  $n$  может быть больше количества элементов в строке или столбце.

**Вариант 14**

Осуществить циклический сдвиг элементов квадратной матрицы размером  $M \times N$  вправо на  $k$  элементов таким образом: элементы первой строки сдвигаются в последний столбец сверху вниз, из него – в последнюю строку справа налево, из нее – в первый столбец снизу вверх, из него – в первую строку; для остальных элементов – аналогично.

**Вариант 15**

Дана целочисленная прямоугольная матрица. Определить номер первого из столбцов, содержащих хотя бы один нулевой элемент. Характеристикой строки целочисленной матрицы назовем сумму ее отрицательных четных элементов. Переставляя строки заданной матрицы, расположить их в соответствии с убыванием характеристик.

**Пример выполнения:**

Дана целочисленная прямоугольная матрица. Определить:

- количество отрицательных элементов в тех строках, которые содержат хотя бы один нулевой элемент;
- номера строк и столбцов всех седловых точек матрицы.

```
#include <stdio.h> // для printf() и scanf()
#include <stdlib.h> // для rand() и srand()
#include <math.h>   // для abs()
#include <time.h>   // для time()

int main() {
    int N,M;
    int m[100][100]; // матрица 100 на 100
    // инициализируем счетчик случайных чисел
    srand((unsigned)time(NULL));
    // вводим размер матрицы
    printf("N M=");
    scanf("%i %i", &N, &M);

    // заполняем массив
    printf("Исходная матрица:\n");
    for(int i=0; i<N; i++) {
        for(int j=0; j<M; j++) {
            m[i][j]=rand()%19-9;
            printf("%3i",m[i][j]);
        }
        printf("\n");
    }

    // находим кол-во отрицательных элементов в строках с нулем
    int cnt=0;
    for(int i=0; i<N; i++) {
        bool zero=false;
        for(int j=0; j<M; j++)
            if(m[i][j]==0) zero=true;
        if(zero)
            for(int j=0; j<M; j++) if(m[i][j]<0) cnt++;
    }
    printf("Колво отрицательных в строках с нолями: %i\n",cnt);

    // считаем седловые точки
    int pnts=0;
    for(int i=0; i<N; i++)
        for(int j=0; j<M; j++) {
```

```
    bool min=true,max=true;
    for(int k=0; k<N; k++)
        if(m[k][j]>m[i][j]) max=false;
    for(int k=0; k<M; k++)
        if(m[i][k]<m[i][j]) min=false;
    if(min && max) pnts++;
}
printf("Седловых точек: %i\n",pnts);

return 0;
}
```