## Лабораторная работа №9

## Динамические массивы

***Цель лабораторной работы:*** *изучение структурной организации динамических массивов и способов доступа к их элементам с использованием указателей; совершенствование навыков процедурного программирования на языке C/C++ при решении задач обработки динамических массивов.*

***Задание на программирование:*** *используя технологию процедурного программирования, разработать программу обработки одномерных и двумерных (матриц) динамических массивов в соответствии с индивидуальным заданием.*

***Порядок выполнения работы:***

1. Получить у преподавателя индивидуальное задание и выполнить постановку для каждой из задач: сформулировать условие, определить входные и выходные данные, их ограничения.

2. Разработать математические модели для каждой из задач: описать с помощью формул и рисунков структуру массивов и процесс их преобразования.

3. Провести структуризацию каждой из задач. С этой целью выделить в них подзадачи, которые будут реализованы в виде отдельных функций (ввод исходных данных, вывод содержимого массива, обработка массива в соответствии с заданием и т.д.). При этом запрещается совмещать в одной функции решение нескольких подзадач.

4. Обязательное требование – введение собственных типов данных, логически точно разделяющих и группирующих информацию, используемую при решении конкретной задачи (тип значения элемента массива, тип массива и т.п.).

5. Построить схемы алгоритмов основных функций.

6. Составить программу на языке *C*/*C*++.

7. Входные данные на этапах тестирования и демонстрации работы преподавателю должны задаваться либо с использованием специально подобранных арифметических формул, либо вводиться с клавиатуры по запросу. **Датчики псевдослучайных чисел использовать запрещается**.

Если это не оговорено в конкретном варианте задания, при тестировании значения элементов в каждом исходном массиве **должны быть разными**.

8. Выходные данные должны выводиться на экран с пояснениями. Операторы вывода результатов работы должны находиться либо в функции *main*(), либо в специальной функции вывода (например, преобразованного массива), вызов которой осуществляется из функции *main*().

9. Проверить и продемонстрировать преподавателю работу программы на полном наборе тестов, в том числе с ошибочными входными данными. Входные и выходные массивы должны выводиться в одном и том же формате.

10. Использовать стандартные потоковые объекты ввода/вывода ***cin*** и ***cout***.

11. Оформить отчет о лабораторной работе в составе: постановка задачи, математическая модель, схемы алгоритмов основных функций, текст программы, контрольные примеры (скриншоты).

12. Текст программы в отчете не должен представлять из себя скриншот.

13. Скриншоты тестов должны легко читаться. Все их неинформативные части должны быть удалены.

***Варианты индивидуальных заданий***

**1** **2**

**8 3**

**7 4**

**6 5**

**1.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить сумму значений элементов массива, лежащих между первым элементом с наименьшим значением и последним элементом массива с положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области 2 (см. рисунок).

**2.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить произведение значений элементов массива с нечётными номерами, лежащих между первым элементом с наибольшим по абсолютной величине значением и первым элементом массива с положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области 3 (см. рисунок).

**3.**

А. Дан массив *z*0, *z*1, *z*2,…, *zn-*1. Определить абсолютную величину суммы значений элементов массива, лежащих между первым элементом с наибольшим значением и последним элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области 4 (см. рисунок).

**4.**

А. Дан массив *a*0, *a*1, *a*2,…, *an-*1. Определить абсолютную величину произведения значений элементов массива с чётными номерами, лежащих между последним элементом с наименьшим значением и первым элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области 6 (см. рисунок).

**5.**

А. Дан массив *b*0, *b*1, *b*2,…, *b*2*n-*1. Определить сумму абсолютных значений элементов массива, лежащих между первым элементом с наибольшим значением и первым элементом массива с отрицательным значением, имеющим номер больше, чем *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области 7 (см. рисунок).

**6.**

А. Дан массив *c*0, *c*1, *c*2,…, *cn-*1. Определить произведение абсолютных значений элементов массива с чётными номерами, лежащих между первым элементом массива с отрицательным значением и последним элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области 8 (см. рисунок).

**7.**

А. Дан массив *d*0, *d*1, *d*2,…, *d*2*n-*1. Определить сумму абсолютных значений элементов массива, лежащих между первым элементом с наименьшим неотрицательным значением и последним элементом массива с отрицательным значением, имеющим номер меньше *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [1+2+3] (см. рисунок).

**8.**

А. Дан массив *x*0, *x*1, *x*2,…, *x*2*n-*1. Определить произведение абсолютных значений элементов массива, лежащих между последним элементом с наименьшим значением и первым элементом массива с положительным значением, имеющим номер больше, чем *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области [2+3+4] (см. рисунок).

**9.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить абсолютную величину суммы значений элементов массива с чётными номерами, лежащих между первым элементом с наибольшим значением и последним элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [3+4+5] (см. рисунок).

**10.**

А. Дан массив *z*0, *z*1, *z*2,…, *zn-*1. Определить абсолютную величину произведения значений элементов массива с нечётными номерами, лежащих между последним элементом с наименьшим по абсолютной величине значением и первым элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области [4+5+6] (см. рисунок).

**11.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить количество и произведение отрицательных значений элементов массива, лежащих между первым элементом с наименьшим по абсолютной величине значением и первым элементом с наибольшим значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов столбцов с одинаковыми номерами областей 7 и 8 (см. рисунок).

**12.**

А. Дан массив *z*0, *z*1, *z*2,…, *zn-*1. Определить количество и сумму положительных значений элементов массива с чётными номерами, лежащих между первым элементом с наименьшим значением и последним элементом с наибольшим значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк с одинаковыми номерами областей 5 и 6 (см. рисунок).

**13.**

А. Дан массив *a*0, *a*1, *a*2,…, *a*2*n-*1. Определить произведение положительных значений элементов массива, лежащих между первым элементом с наименьшим по абсолютной величине значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк с одинаковыми номерами областей 3 и 8 (см. рисунок).

**14.**

А. Дан массив *b*0, *b*1, *b*2,…, *b*2*n-*1. Определить сумму абсолютных значений элементов массива, лежащих между первым элементом с наименьшим неотрицательным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк с одинаковыми номерами областей 4 и 7 (см. рисунок).

**15.**

А. Дан массив *c*0, *c*1, *c*2,…, *c*2*n-*1. Определить произведение абсолютных значений отрицательных элементов массива, лежащих между первым элементом с наибольшим неположительным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов столбцов с одинаковыми номерами областей 1 и 6 (см. рисунок).

**16.**

А. Дан массив *d*0, *d*1, *d*2,…, *d*2*n-*1. Определить количество и сумму отрицательных значений элементов массива, лежащих между последним элементом с наименьшим неотрицательным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов столбцов с одинаковыми номерами областей 2 и 5 (см. рисунок).

**17.**

А. Дан массив *x*0, *x*1, *x*2,…, *x*2*n-*1. Определить количество и произведение положительных значений элементов массива, лежащих между первым элементом с наименьшим по абсолютной величине значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 2 и элементов столбцов области 8 (см. рисунок) с одинаковыми номерами.

**18.**

А. Дан массив *y*0, *y*1, *y*2,…, *y*2*n-*1. Определить сумму значений элементов массива с чётными номерами, лежащих между первым элементом с наименьшим неотрицательным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 8 и элементов столбцов области 6 (см. рисунок) с одинаковыми номерами.

**19.**

А. Дан массив *z*0, *z*1, *z*2,…, *z*2*n-*1. Определить произведение значений элементов массива с нечётными номерами, лежащих между первым элементом с наименьшим значением и первым элементом с положительным значением, имеющим номер больше, чем *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области [1+2] и элементов столбцов области [7+8] (см. рисунок) с одинаковыми номерами.

**20.**

А. Дан массив *a*0, *a*1, *a*2,…, *a*2*n-*1. Определить сумму значений элементов массива с чётными номерами, лежащих между первым элементом с наибольшим неположительным значением среди элементов массива и последним элементом с отрицательным значением, имеющим номер меньше, чем *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области [1+2+3] и элементов столбцов области [6+7+8] (см. рисунок) с одинаковыми номерами.

**21.**

А. Дан массив *a*0, *a*1, *a*2,…, *an-*1. Определить абсолютную величину разности значений элементов массива с наибольшим и наименьшим значениями и суммарное количество элементов массива с отрицательными значениями.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области [6+7+8] (см. рисунок).

**22.**

А. Дан массив *b*0, *b*1, *b*2,…, *bn-*1. Определить произведение абсолютных значений элементов массива с чётными номерами, лежащих между первым элементом с наименьшим значением и последним элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [1+2+8] (см. рисунок).

**23.**

А. Дан массив *c*0, *c*1, *c*2,…, *c*2*n-*1. Определить сумму абсолютных значений элементов массива, лежащих между первым элементом с наибольшим неположительным значением и последним элементом массива с отрицательным значением, имеющим номер меньше *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений элементов области [1+2+3+4+5] (см. рисунок).

**24.**

А. Дан массив *d*0, *d*1, *d*2,…, *d*2*n-*1. Определить значение наибольшего по абсолютной величине элемента массива, лежащего между первым элементом с положительным значением и первым элементом массива с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области [1+2+6+7+8] (см. рисунок).

**25.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить значение наименьшего по абсолютной величине элемента массива, лежащего между последним элементом с отрицательным значением и последним элементом массива с положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [3+4+5+6+7] (см. рисунок).

**26.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить сумму абсолютных значений элементов массива с нечётными номерами, лежащих между первым элементом массива с положительным значением и первым элементом с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [1+2+3+4+8] (см. рисунок).

**27.**

А. Дан массив *z*0, *z*1, *z*2,…, *z*2*n-*1. Определить произведение абсолютных значений элементов массива с чётными номерами, лежащих между первым элементом массива с положительным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [5+6+7+8+1] (см. рисунок).

**28.**

А. Дан массив *a*0, *a*1, *a*2,…, *a*2*n-*1. Определить абсолютную величину суммы значений элементов массива с нечётными номерами, лежащих между последним элементом массива с положительным значением и элементом с номером *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области [2+3+4+5+6] (см. рисунок).

**29.**

А. Дан массив *b*0, *b*1, *b*2,…, *b*2*n-*1. Определить абсолютную величину произведения значений элементов массива с нечётными номерами, лежащих между первым элементом массива с отрицательным значением и элементом с номером *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [4+5+6+7+8] (см. рисунок).

**30.**

А. Дан массив *c*0, *c*1, *c*2,…, *c*2*n-*1. Определить сумму абсолютных значений элементов массива с чётными номерами, лежащих между последним элементом массива с отрицательным значением и элементом с номером *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [1+2+3+7+8] (см. рисунок).

**31.**

А. Дан массив *b*0, *b*1, *b*2,…, *b*2*n-*1. Определить произведение значений элементов массива с чётными номерами, лежащих между первым элементом с наибольшим по абсолютной величине значением и последним элементом с отрицательным значением, имеющим номер меньше, чем *n*.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области [5+6+7] и элементов столбцов области [2+3+4] (см. рисунок) с одинаковыми номерами.

**32.**

А. Дан массив *c*0, *c*1, *c*2,…, *c*2*n-*1. Определить количество и сумму абсолютных значений элементов массива с чётными номерами, лежащих между первым элементом с наименьшим неотрицательным значением и первым элементом с отрицательным значением, имеющим номер больше, чем *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 6 и элементов столбцов области 4 (см. рисунок) с одинаковыми номерами.

**33.**

А. Дан массив *d*0, *d*1, *d*2,…, *dn-*1. Определить сумму значений элементов массива с чётными номерами, лежащих между последним элементом массива с положительным значением и последним элементом с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 1 и элементов столбцов области 7 (см. рисунок) с одинаковыми номерами.

**34.**

А. Дан массив *x*0, *x*1, *x*2,…, *x*2*n-*1. Определить абсолютную величину суммы значений элементов массива, лежащих между первым элементом с наименьшим неотрицательным значением и первым элементом с положительным значением, имеющим номер больше, чем *n*-1.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 7 и элементов столбцов области 5 (см. рисунок) с одинаковыми номерами.

**35.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить сумму значений элементов массива с нечётными номерами, лежащих между последним элементом с наименьшим значением и последним элементом массива с положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 5 и элементов столбцов области 3 (см. рисунок) с одинаковыми номерами.

**36.**

А. Дан массив *z*0, *z*1, *z*2,…, *zn-*1. Определить произведение суммы положительных значений элементов массива, лежащих между первым элементом с наибольшим неположительным значением и последним элементом массива с отрицательным значением, на их количество.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 3 и элементов столбцов области 1 (см. рисунок) с одинаковыми номерами.

**37.**

А. Дан массив *a*0, *a*1, *a*2,…, *an-*1. Определить произведение суммы отрицательных значений элементов массива, предшествующих последнему элементу с положительным значением, на общее количество элементов массива с отрицательными значениями.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области 4 и элементов столбцов области 2 (см. рисунок) с одинаковыми номерами.

**38.**

А. Дан массив *b*0, *b*1, *b*2,…, *bn-*1. Определить произведение суммы абсолютных значений элементов массива с чётными номерами, предшествующих последнему элементу массива с отрицательным значением, на общее количество элементов массива с положительными значениями.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк области [5+6] и элементов столбцов области [3+4] (см. рисунок) с одинаковыми номерами.

**39.**

А. Дан массив *c*0, *c*1, *c*2,…, *c*2*m-*1. Написать программу преобразования массива в массив с элементами, соответственно равными: *c*2*m-*1, *c*2*m*-2,…, *cm*, *c*0, *c*1, …, *cm-*1. Дополнительные массивы не использовать.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области [5+6+7] (см. рисунок).

**40.**

А. Дан массив *d*0, *d*1, *d*2,…, *d*3*m-*1. Написать программу построения нового массива с элементами, равными: *d*0, *d*3, *d*6, …, *d*3*m*-3, *d*1, *d*4, …, *d*3*m-*2, *d*2, *d*5, …, *d*3*m-*1.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее геометрическое положительных значений элементов области [8+1+2] (см. рисунок).

**41.**

А. Дан массив *b*0, *b*1, *b*2,…, *bn-*1. Определить сумму положительных значений всех его элементов и количество элементов с отрицательными значениями, лежащих между первым элементом с наименьшим по абсолютной величине значением и первым элементом с положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений отрицательных элементов области 1 (см. рисунок).

**42.**

А. Дан массив *a*0, *a*1, *a*2,…, *an-*1. Определить произведение положительных значений его элементов, лежащих между первым элементом с наибольшим неположительным значением и первым элементом с наименьшим положительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее геометрическое положительных значений элементов области 2 (см. рисунок).

**43.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить сумму значений всех его элементов, лежащих между первым элементом с наименьшим неотрицательным значением и первым элементом с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее геометрическое положительных значений элементов области 3 (см. рисунок).

**44.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить сумму наибольшего и наименьшего значений элементов массива и общее количество элементов массива, имеющих отрицательные значения.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти сумму абсолютных значений отрицательных элементов области 4 (см. рисунок).

**45.**

А. Дан массив *y*0, *y*1, *y*2,…, *yn-*1. Определить произведение наибольшего по абсолютной величине и наименьшего по абсолютной величине значений элементов массива и сумму неположительных значений элементов массива.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее арифметическое значений положительных элементов области 5 (см. рисунок).

**46.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить произведение положительных значений элементов массива с чётными номерами и сумму неположительных значений элементов с нечётными номерами.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти среднее геометрическое значений положительных элементов области 6 (см. рисунок).

**47.**

А. Дан массив *c*0, *c*1, *c*2,…, *cn-*1. Определить произведение абсолютных значений элементов массива с чётными номерами и сумму значений элементов с нечётными номерами, номера которых больше номера первого элемента с наибольшим неположительным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти сумму отрицательных значений элементов области 7 (см. рисунок), умноженную на их количество.

**48.**

А. Дан массив *b*0, *b*1, *b*2,…, *bn-*1. Определить произведение суммы неположительных значений элементов массива, лежащих между первым элементом с наибольшим значением и первым элементом с наименьшим значением, на их количество.

Б. В заданной квадратной матрице размера 2*n×*2*n* найти сумму абсолютных значений элементов области 8 (см. рисунок), умноженную на количество элементов с отрицательными значениями, принадлежащих этой области.

**49.**

А. Дан массив *a*0, *a*1, *a*2,…, *an-*1. Определить количество и произведение значений элементов массива с чётными номерами, лежащих между первым элементом с наибольшим по абсолютной величине значением и последним элементом с отрицательным значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов строк с одинаковыми номерами областей 1 и 2 (см. рисунок).

**50.**

А. Дан массив *x*0, *x*1, *x*2,…, *xn-*1. Определить произведение положительных значений элементов массива с нечётными номерами, лежащих между первым элементом с наименьшим по абсолютной величине значением и первым элементом с наименьшим значением.

Б. В заданной квадратной матрице размера 2*n×*2*n* поменять местами значения элементов столбцов с одинаковыми номерами областей 3 и 4 (см. рисунок).

***Пример 1 решения задачи на обработку одномерного массива для варианта задания вида:***

Дан массив *a*0, *a*1,…, *an-*1. Найти и вывести номер элемента этого массива, для которого сумма абсолютных величин разностей со значениями соседних элементов максимальна. Для крайних элементов использовать циклическое замыкание.

Математическая модель решения и схемы алгоритмов аналогичны *примеру* 2 на обработку статических одномерных массивов.

***Математическая модель решения***

Решение задачи начинается с ввода исходных данных. Прежде всего, необходимо ввести значение (*n*) размера массива.

После этого необходимо задать значения всех элементов массива. Доступ к элементам массива осуществляется по их индексу (номеру данного элемента массива). Поэтому перебираем все индексы элементов массива от 0 до *n*-1 (нумерация элементов массива начинается с 0) и задаём значения элементов с текущими номерами.

Пусть исходный массив имеет вид (*m*=10):

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** |  |  |  |  |  |  | ***n*-2** | ***n*-1** |
| **4** | **-3** | **7** | **2** | **-5** | **6** | **1** | **-2** | **3** | **5** |

Теперь можно приступать к решению задачи. Вначале определяем сумму абсолютных величин разностей со значениями соседних элементов для элемента массива с индексом 0. Поскольку этот элемент массива является крайним, по условию задачи используем циклическое замыкание элементов массива и определяем величину *max*=*fabs*(*a*[*n*-1]-*a*[0])+*fabs*(*a*[1]-*a*[0]). Эту величину принимаем за исходное значение искомой суммы и фиксируем индекс этого элемента *imax*=0.

Далее, перебирая индексы элементов массива с 1 до *n*-1, для остальных элементов вычисляем сумму абсолютных величин разностей их значений со значениями соседних элементов и сравниваем эти значения со значением *max*. Если значение вычисленной суммы превышает текущее значение *max*, меняем значение *max* и запоминаем индекс этого элемента *imax*. При этом для элемента массива с индексом *m*-1 используем циклическое замыкание элементов массива и искомую величину суммы определяем как *fabs*(*a*[0]*-a*[*n*-1])*+fabs*(*a*[*n*-2]-*a*[*n*-1]).

В результате индекс элемента массива, для которого сумма абсолютных величин разностей со значениями соседних элементов максимальна, зафиксирован в переменной *imax,* а сама величина суммы зафиксирована в переменной *max*.

Для рассматриваемого примера индекс искомого элемента *imax*=4, при этом сумма абсолютных величин разностей со значениями соседних элементов *max*=18:

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** | **2** | **3** | **4** | ***5*** | **6** | **7** | **8** | **9** |
| **4** | **-3** | **7** | **2** | **-5** | **6** | **1** | **-2** | **3** | **5** |
|  |  |  |  | ***imax*** |  |  |  |  |  |

Задача решена.

***Схема алгоритма поиска индекса элемента массива (функция nomer)***

*nomer*(*a*, &*max*, *n*)

*imax*=0

*max*=*fabs*(*a*[*n*-1]-*a*[0])+*fabs*(*a*[1]-*a*[0])

*i*=1; *i*<*n*-1; *i*++

*pr=fabs*(*a*[*i*-1]*-a*[*i*])*+fabs*(*a*[*i*+1]-*a*[*i*])

нет

*max*<*pr*

да

*imax*=*i*

*max*=*pr*

*pr=fabs*(*a*[0]*-a*[*n*-1])*+fabs*(*a*[*n*-2]-*a*[*n*-1])

*max*<*pr*

нет

да

*imax*=*n*-1

*max*=*pr*

*return*(*imax*)

***Текст программы (пример 1)***

/\*Массивы динамические одномерные

Найти номер элемента числового массива, для которого сумма

разностей со значениями соседних элементов максимальна.

Для крайних элементов использовать циклическое замыкание.\*/

#include<iostream.h>

#include<math.h>

#include<locale.h>

using namespace std;

typedef int telem ; //объявление типа элемента массива

typedef telem \*tmas ; //объявление типа "указатель на telem"

int nomer(tmas a, telem &max, int n) ;

void inputmas(tmas a, int n) ;

//main\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main()

{int n ; //размер массива

setlocale(LC\_ALL,"Russian") ;

cout << "\n Найти номер элемента массива, для которого сумма" << endl ;

cout << " разностей со значениями соседних элементов максимальна" ;

//Ввод исходных данных

cout << "\n Введите количество элементов массива: " ;

cin >> n ;

tmas a = new telem [n] ; /\*создание переменной-указателя на telem,

выделение динамической памяти под массив

адрес начала области заносится в a\*/

int nom ; //номер искомого элемента

telem max ; //значение максимальной разности

inputmas(a, n) ;

//Поиск номера элемента

nom = nomer(a, max, n) ;

cout << " Искомый номер элемента массива: " << nom << endl ;

cout << " Значение элемента: " << \*(a+nom) << endl ;

cout << " Сумма разностей = " << max ;

delete [] a ; //освобождение динамической памяти

return 0 ;

}

//main\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//nomer\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int nomer(tmas a, telem &max, int n)

{telem pr ; //текущее значение разности

int imax = 0 ; //за максимум принимаем первый по счету элемент

max = fabs(\*(a+n-1) - \*a) + fabs(\*(a+1) - \*a) ;

for(int i = 1 ; i < n - 1 ; i++)

if(max < (pr = fabs(\*(a+i-1) - \*(a+i)) + fabs(\*(a+i+1) - \*(a+i))))

{imax = i ;

max = pr ;

}

if(max < (pr = fabs(\*a - \*(a+n-1)) + fabs(\*(a+n-2) - \*(a+n-1))))

{imax = n - 1 ;

max = pr ;

}

return imax ;

}

//nomer\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//inputmas\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void inputmas(tmas a, int n)

{cout << " Введите одной строкой элементы массива из " ;

cout << n << " чисел и нажмите <Enter>" << endl << " " ;

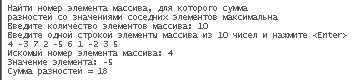
for(int i = 0 ; i < n ; i++)

cin >> \*(a+i) ;

}

//inputmas\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Скриншот результатов выполнения программы (пример 1)***



***Пример 2 решения задачи для варианта задания вида:***

В заданной матрице найти строку с максимальной суммой значений элементов.

Математическая модель решения и схемы алгоритмов аналогичны *примеру* 2 на обработку статических двумерных массивов.

***Математическая модель решения***

Решение задачи начинается с ввода исходных данных. Из формулировки задачи понятно, что исходная матрица имеет произвольное число строк (*str*) и столбцов (*sto*). Следовательно, прежде всего, необходимо ввести значения (*str, sto*) размеров матрицы.

После этого необходимо задать значения всех элементов матрицы. Доступ к элементам двумерного массива (матрицы) осуществляется по двум индексам: номеру строки и номеру столбца, на пересечении которых находится данный элемент массива. Поэтому перебираем все строки (с 0-ой до *str*-1). Внутри каждой строки перебираем все столбцы (с 0-ого до *sto*-1) и задаём значения элементов, лежащих на пересечении строки и столбца с текущими номерами.

Например, исходная матрица имеет вид (*str*=6, *sto*=10, строка с номером 0 выделена цветом):

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** |  |  |  |  |  |  | ***sto*-1** |
| **0** | **1** | **3** | **5** | **4** | **7** | **2** | **8** | **1** | **6** | **2** |
| **1** | **3** | **5** | **1** | **6** | **2** | **7** | **8** | **4** | **5** | **4** |
| **2** | **5** | **6** | **4** | **5** | **6** | **4** | **5** | **5** | **5** | **5** |
|  | **1** | **2** | **5** | **2** | **3** | **1** | **4** | **5** | **2** | **3** |
|  | **4** | **4** | **1** | **2** | **1** | **5** | **1** | **2** | **3** | **1** |
| ***str*-1** | **9** | **1** | **1** | **1** | **4** | **2** | **1** | **4** | **2** | **2** |

Теперь можно приступать к решению задачи. Вначале определяем сумму значений элементов, принадлежащих нулевой строке (в приведённом примере сумма равна 39). Эту величину принимаем за исходное значение максимальной суммы значений элементов, принадлежащих одной строке.

Затем перебираем номера строк матрицы с 1 до *str*-1. Для каждой строки вычисляем сумму значений её элементов и сравниваем с текущим значением максимальной суммы. Если в строке сумма значений элементов превышает текущее значение максимальной суммы, меняем значение текущего максимума и фиксируем номер строки, где найдена эта сумма. В результате для рассматриваемого примера определяем, что максимальная сумма значений элементов (50) содержится в строке с номером 2 (выделена цветом).

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** | **2** |  |  |  |  |  |  | ***sto*-1** |
| **0** | **1** | **3** | **5** | **4** | **7** | **2** | **8** | **1** | **6** | **2** |
| **1** | **3** | **5** | **1** | **6** | **2** | **7** | **8** | **4** | **5** | **4** |
| **2** | **5** | **6** | **4** | **5** | **6** | **4** | **5** | **5** | **5** | **5** |
|  | **1** | **2** | **5** | **2** | **3** | **1** | **4** | **5** | **2** | **3** |
|  | **4** | **4** | **1** | **2** | **1** | **5** | **1** | **2** | **3** | **1** |
| ***str*-1** | **9** | **1** | **1** | **1** | **4** | **2** | **1** | **4** | **2** | **2** |

Задача решена.

***Схема алгоритма поиска строки с максимальной суммой значений элементов (функция poisk\_str)***

*poisk\_str*(*a*, *str*, *sto*, &*imax*)

*j*=0; *j*<*sto*; *j*++

*max*+=a[0][*j*]

*return(max)*

*max*=0

*pr*=0

*max*<*pr*

*imax*=0

*imax*=*i*

*max*=*pr*

*i*=1; *i*<*str*; *i*++

*j*=0; *j*<*sto*; *j*++

*pr*+=*a*[*i*][*j*]

да

нет

***Текст программы (пример 2)***

/\*Массивы динамические двумерные.

В заданной матрице найти строку с максимальной суммой значений элементов\*/

#include<iostream.h>

#include<locale.h>

using namespace std;

typedef int telem ; //определение типа элементов массива

typedef telem \*tstr ; //определение типа "указатель на telem"

typedef tstr \*tmatr; //тип "указатель на указатель на telem"

void input\_matr(tmatr a, int str, int sto) ;

void output\_matr(tmatr, int, int) ;

telem poisk\_str(tmatr, int, int, int&) ;

//main\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main()

{int imax, //номер строки с максимальной суммой элементов

str, //число строк матрицы

sto ; //число столбцов матрицы

tmatr a ; //переменная-указатель на указатель на telem

telem max ; //максимальная сумма элементов

setlocale(LC\_ALL,"Russian") ;

cout << " Найти номер строки двумерного числового массива,\n" ;

cout << " содержащей максимальную сумму значений элементов\n" ;

cout << " Введите количество строк и столбцов матрицы: " ;

cin >> str >> sto;

a = new tstr[str] ; /\*выделение динамической памяти под массив

указателей на строки массива\*/

for(int i = 0; i < str; i++) //выделение памяти под каждую строку

\*(a+i) = new telem[sto] ; /\*каждому элементу массива указателей

на строки присваивается адрес начала

области памяти, выделяемой под строку\*/

Input\_matr(a, str, sto) ;

cout << " Исходная матрица:\n" ;

output\_matr(a, str, sto) ;

max = poisk\_str(a, str, sto, imax) ;

cout << " Максимальная сумма значений элементов строки=" << max ;

cout << "\n Она содержится в строке с номером " << imax << endl ;

cout << endl << " Для завершения нажмите <Enter>" ;

//Освобождение динамической памяти

for(int i = 0; i < str; i++)

delete \*(a + i) ;

delete []a ;

return 0 ;

}

//main\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//input\_matr\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void input\_matr(tmatr a, int str, int sto)

{cout << " Введите построчно через пробел элементы" ;

cout << " массива размера " << str << "x" << sto ;

cout << "\n После ввода строки нажимайте <Enter>" << endl ;

for(int i = 0; i < str; i++)

for(int j = 0; j < sto; j++)

cin >> \*(\*(a + i) + j) ;

}

//input\_matr\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//poisk\_str\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

telem poisk\_str(tmatr a, int str, int sto, int &imax)

{telem max, //максимальная сумма значений элементов

pr ; //сумма значений элементов текукущей строки

imax = 0 ; //предполагаем, что макс сумма элементов в 0-ой строке

max = 0 ;

for(int j = 0; j < sto; j++)

max += \*(\*a + j) ;

for(int i = 1; i < str; i++)

{pr = 0 ;

for(j = 0; j < sto; j++)

pr += \*(\*(a + i) + j) ;

if(max < pr)

{imax = i ;

max = pr ;

}

}

return max ;

}

//poisk\_str\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//output\_matr\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void output\_matr(tmatr a, int str, int sto)

{int i, j ;

for(i = 0; i < str; i++)

{for(j = 0; j < sto; j++)

{cout.width(3) ; //ширина поля выводимого параметра

cout << \*(\*(a + i) + j) << ' ' ;

}

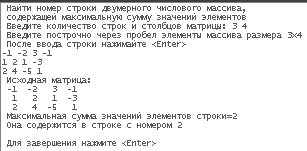
cout << '\n';

}

}

//output\_matr\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Скриншот результатов выполнения программы (пример 2)***



***Пример 3 решения задачи для варианта задания вида:***

В квадратной матрице размера 2*n*×2*n* поменять местами значения элементов строк области [1+2+3+4] и значения элементов столбцов области [8+7+6+5] с одинаковыми номерами (см. рисунок).

***Математическая модель решения***

Решение задачи начинается с ввода исходных данных. Из формулировки задачи понятно, что исходная матрица квадратная с **чётным** числом (2*n*) строк и столбцов. Следовательно, прежде всего, необходимо ввести значение (*n*) половины размера квадратной матрицы.

После этого необходимо задать значения всех элементов матрицы. Доступ к элементам двумерного массива (матрицы) осуществляется по двум индексам: номеру строки и номеру столбца, на пересечении которых находится данный элемент массива (нумерация строк и столбцов начинается с 0). Поэтому перебираем все строки (с 0-ой до 2*n*-1). Внутри каждой строки перебираем все столбцы (с 0-ого до 2*n*-1) и задаём значения элементов, лежащих на пересечении строки и столбца с текущими номерами.

Пусть в качестве примера исходная матрица имеет вид (*n*=5):

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** |  |  | **n-1** | **n** |  |  | **2n-2** | **2n-1** |
| **0** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** |
| **1** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** | **19** |
|  | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** |
|  | **30** | **31** | **32** | **33** | **34** | **35** | **36** | **37** | **38** | **39** |
| **n-1** | **40** | **41** | **42** | **43** | **44** | **45** | **46** | **47** | **48** | **49** |
| **n** | **50** | **51** | **52** | **53** | **54** | **55** | **56** | **57** | **58** | **59** |
|  | **60** | **61** | **62** | **63** | **64** | **65** | **66** | **67** | **68** | **69** |
|  | **70** | **71** | **72** | **73** | **74** | **75** | **76** | **77** | **78** | **79** |
| **2n-2** | **80** | **81** | **82** | **83** | **84** | **85** | **86** | **87** | **88** | **89** |
| **2n-1** | **90** | **91** | **92** | **93** | **94** | **95** | **96** | **97** | **98** | **99** |

Теперь можно приступать к решению задачи. Требуется поменять местами значения элементов строк области [1+2+3+4] (т.е. значения элементов строк, лежащие выше главной диагонали) и значения элементов столбцов области [8+7+6+5] (т.е. значения элементов столбцов, лежащие ниже главной диагонали) с одинаковыми номерами. Значения элементов 0-ой строки меняются со значениями 0-го столбца и т.д. При таком обмене значения элементов, принадлежащих главной диагонали, останутся на своём месте.

Поэтому в обмене участвуют значения элементов строк с 0 до 2*n*-2 и элементы столбцов с 0 до 2*n*-2. Используем метод «трёх стаканов». Перебираем номера строк с 0 до 2*n*-2. По очереди засылаем значения элементов текущей *i*-ой строки с индексами *a*[*i*][*j*+1] (где *j* – текущий номер элемента в строке *i*, который меняется с номера *i* до 2*n*-2) во вспомогательную переменную *z*. На их место записываем значения элементов столбца с индексами *a*[*j*+1][*i*], а затем на их место записываем значения из переменной *z*.

В результате такого обмена получаем преобразованную матрицу:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **0** | **1** |  |  | **n-1** | **n** |  |  | **2n-2** | **2n-1** |
| **0** | **0** | **10** | **20** | **30** | **40** | **50** | **60** | **70** | **80** | **90** |
| **1** | **1** | **11** | **21** | **31** | **41** | **51** | **61** | **71** | **81** | **91** |
|  | **2** | **12** | **22** | **32** | **42** | **52** | **62** | **72** | **82** | **92** |
|  | **3** | **13** | **23** | **33** | **43** | **53** | **63** | **73** | **83** | **93** |
| **n-1** | **4** | **14** | **24** | **34** | **44** | **54** | **64** | **74** | **84** | **94** |
| **n** | **5** | **15** | **25** | **35** | **45** | **55** | **65** | **75** | **85** | **95** |
|  | **6** | **16** | **26** | **36** | **46** | **56** | **66** | **76** | **86** | **96** |
|  | **7** | **17** | **27** | **37** | **47** | **57** | **67** | **77** | **87** | **97** |
| **2n-2** | **8** | **18** | **28** | **38** | **48** | **58** | **68** | **78** | **88** | **98** |
| **2n-1** | **9** | **19** | **29** | **39** | **49** | **59** | **69** | **79** | **89** | **99** |

Задача решена.

***Схема алгоритма обмена значений элементов строк области [1+2+3+4] и элементов столбцов области [8+7+6+5] (функция obmen)***

*obmen*(*a*, *n*)

*i*=0; *i*<*n*-1; *i*++

*j*=*i*; *j*<*n*-1; *j*++

*z*= *a*[*i*][*j*+1]

*a*[*i*][*j*+1]= *a*[*i*+1][*j*]

*a*[*i*][*j*]=*z*

*return*

***Текст программы (пример 3)***

/\*Массивы динамические двумерные.

В квадратной матрице размера 2nx2n поменять местами значения элементов

строк области [1+2+3+4] и значения элементов столбцов

области [8+7+6+5] с одинаковыми номерами (см. рисунок)\*/

#include<iostream.h>

#include<locale.h>

using namespace std;

typedef int telem ; //определение типа элементов массива

typedef telem \*tstr ; //определение типа "указатель на telem"

typedef tstr \*tmatr; //тип "указатель на указатель на telem"

void input\_matr(tmatr, int) ;

void output\_matr(tmatr, int) ;

void obmen(tmatr, int) ;

//main\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int main()

{int n ; //число строк (столбцов) матрицы

tmatr a ; //переменная-указатель на указатель на telem

setlocale(LC\_ALL,"Russian") ;

cout << " В квадратной матрице размера 2nx2n поменять местами" ;

cout << " строки\n области [1+2+3+4] и столбцы" ;

cout << " области [8+7+6+5]\n с одинаковыми номерами.\n" ;

cout << " Введите половину размера матрицы (не больше 4): n = " ;

cin >> n ;

a = new tstr[2\*n] ; /\*выделение динамической памяти под массив

указателей на строки массива\*/

for(int i = 0; i < 2\*n; i++) //выделение памяти под каждую строку:

\*(a+i) = new telem[2\*n] ; /\*каждому элементу массива указателей

на строки присваивается адрес начала

области памяти, выделяемой под строку\*/

input\_matr(a, 2\*n) ;

cout << " Исходная матрица:\n" ;

output\_matr(a, 2\*n) ;

obmen(a, 2\*n) ;

cout << " Результирующая матрица:\n" ;

output\_matr(a, 2\*n) ;

cout << endl << " Для завершения нажмите <Enter>" ;

//Освобождение динамической памяти

for(int i = 0; i < n; i++)

delete \*(a + i) ;

delete []a ;

return 0 ;

}

//main\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//input\_matr\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void input\_matr(tmatr a, int n)

{cout << " Значения элементов" ;

cout << " массива размера " << n << "x" << n ;

cout << "\n при тестировании вводятся автоматически:" << endl ;

for(int i = 0; i < n; i++)

for(int j = 0; j < n; j++)

\*(\*(a+i)+j) = 10 \* i + j ;

}

//input\_matr\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//output\_matr\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void outputmatr(tmatr a, int n)

{for(int i = 0; i < n; i++)

{for(int j = 0; j < n; j++)

{cout.width(3) ; //ширина поля выводимого параметра

cout << \*(\*(a+i)+j) << ' ' ;

}

cout << '\n';

}

}

//output\_matr\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//obmen\_beg\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void obmen(tmatr a, int n)

{telem z ;

for(int i = 0; i < n-1; i++)

for(int j = i; j < n-1; j++)

{z = \*(\*(a+i)+j+1) ;

\*(\*(a+i)+j+1) = \*(\*(a+j+1)+i) ;

\*(\*(a+j+1)+i) = z ;

}

}

//obmen\_end\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Скриншот результатов выполнения программы (пример 3)***

