*Федеральное государственное бюджетное образовательное учреждениевысшего профессионального образования*

|  |  |
| --- | --- |
|  | ***«Московский государственный технический университет***  ***имени Н.Э. Баумана»***  ***(МГТУ им. Н.Э. Баумана)*** |

ФАКУЛЬТЕТ: Информатики и систем управления

КАФЕДРА: Компьютерные системы и сети

**Отчет**

**по усложнённой лабораторной работе №** 2

**Название лабораторной работы:** Работа с массивами С++.

**Дисциплина:** Объектно-ориентированное программирование

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент гр. ИУ6-22 |  |  | | Боярских Никита |
|  | (Подпись, дата) | |  | (И. О. Фамилия) |
| Преподаватель |  | |  | Черноусова Т. Г. |
|  | (Подпись, дата) | |  | (И. О. Фамилия) |

**Вариант 1**

Москва, 2017

**Задание:**

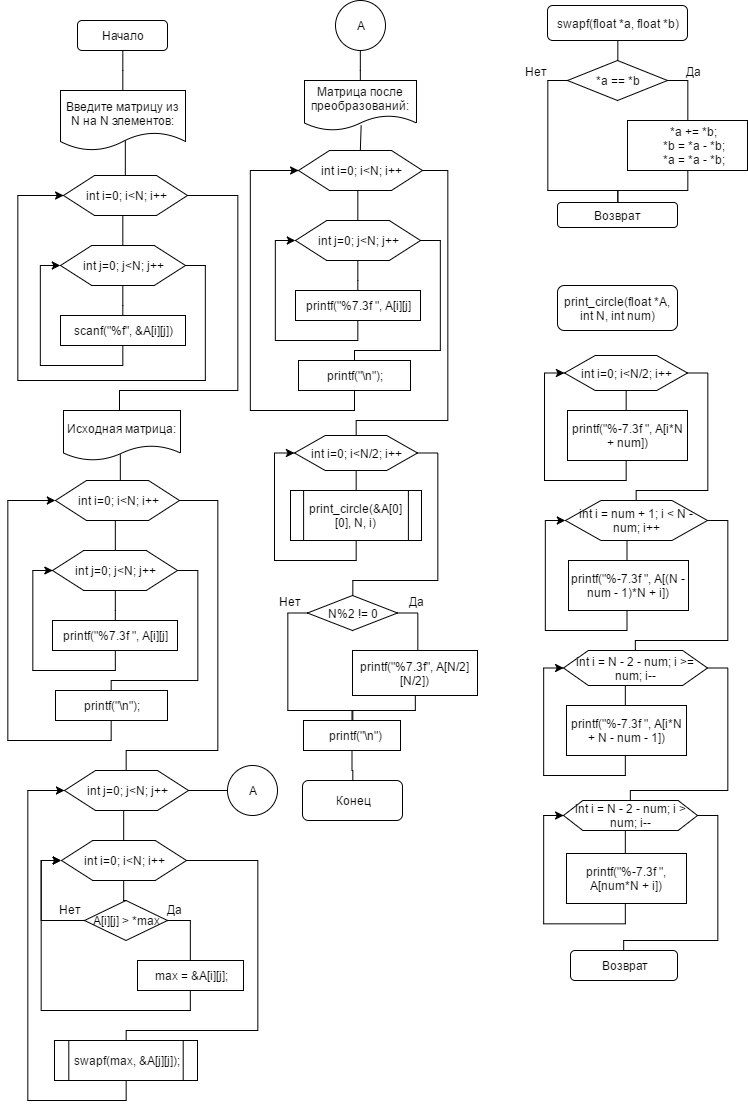
В квадратной матрице А(*N,N*), содержащей вещественные элементы, в каждом столбце поменять местами максимальный элемент с диагональным. Распечатать:

а) исходную и преобразованную матрицы;

б) в порядке спирали против часовой стрелки все элемента начиная с первого.

Для поставленной задачи написан алгоритм

(схема алгоритма представлена на рис. 1)



*Рисунок 1 - схема алгоритма программы, работающей с матрицей*

**Код программы, создающей список строк и выводящей его на экран**

#include <stdio.h>

// Выводит num-ый круг матрицы А размером N\*N

void print\_circle(float \*A, int N, int num) {

for(int i = num; i < N - num; i++) printf("%-7.3f ", A[i\*N + num]); // Вывод левой стороны

for(int i = num + 1; i < N - num; i++) printf("%-7.3f ", A[(N - num - 1)\*N + i]); // Вывод нижней стороны

for(int i = N - 2 - num; i >= num; i--) printf("%-7.3f ", A[i\*N + N - num - 1]); // Вывод правой стороны

for(int i = N - 2 - num; i > num; i--) printf("%-7.3f ", A[num\*N + i]); // Вывод верхней стороны

}

// Меняет значения 2-ух float переменных

void swapf(float \*a, float \*b) {

if(\*a == \*b) return;

\*a += \*b;

\*b = \*a - \*b;

\*a = \*a - \*b;

}

int main(int argc, char \*\*argv) {

int N;

printf("Введите размер N матрицы A(N,N): ");

scanf("%d", &N);

float A[N][N];

printf("Введите матрицу из %d на %d элементов:\n", N, N);

// Ввод матрицы

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

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

scanf("%f", &A[i][j]);

printf("\nИсходная матрица:\n");

// Вывод первоначальной матрицы

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

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

printf("%7.3f ", A[i][j]);

printf("\n");

}

// Ищем максимальный элемент в каждом столбце и меняем с диагональным

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

float \*max = &A[0][j];

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

if(A[i][j] > \*max)

max = &A[i][j];

swapf(max, &A[j][j]);

}

printf("\nМатрица после преобразований:\n");

// Вывод преобразованной матрицы

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

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

printf("%7.3f ", A[i][j]);

printf("\n");

}

printf("\nВывод матрица спиралью против часовой стрелки с первого:\n");

// Вывод матрицы спиралью против часовой стрелки

for(int i=0; i<N/2; i++) print\_circle(&A[0][0], N, i);

// Вывод центрального элемента, если матрица состоит из нечётного количества элементов

if(N%2 != 0) printf("%7.3f", A[N/2][N/2]);

printf("\n")

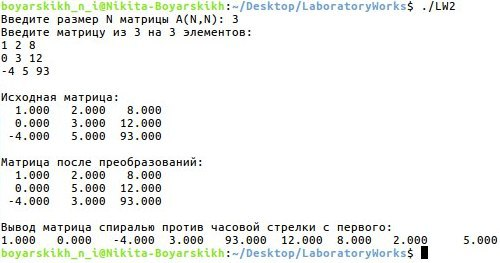
return 0;

}

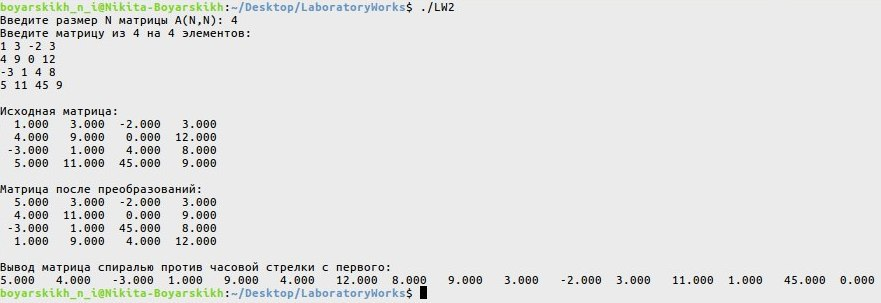
Проведено тестирование программы (см. табл. 1)

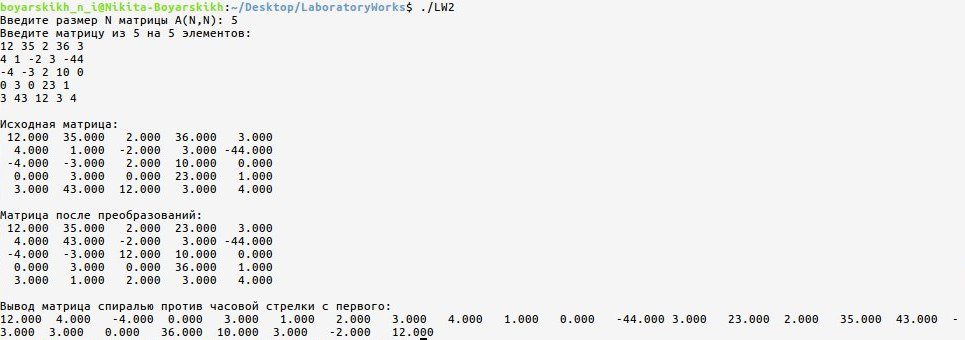
Таблица 1 – результаты тестирования программы, работающей со списком строк

|  |  |  |  |
| --- | --- | --- | --- |
| № | Исходные данные | Ожидаемый результат | Полученный результат |
| 1 | 3  1 2 8  0 3 12  -4 5 93 | 1 2 8  0 5 12  -4 3 93 | 1 2 8  0 5 12  -4 3 93 |
| 2 | 4  1 3 -2 3  4 9 0 12  -3 1 4 8  5 11 45 9 | 5 3 -2 3  4 11 0 9  -3 1 45 8  1 9 4 12 | 5 3 -2 3  4 11 0 9  -3 1 45 8  1 9 4 12 |
| 3 | 5  12 35 2 36 3  4 1 -2 3 -44  -4 -3 2 10 0  0 3 0 23 1  3 43 12 3 4 | 12 35 2 23 3  4 43 -2 3 -44  -4 -3 12 10 0  0 3 0 36 1  3 1 2 3 4 | 12 35 2 23 3  4 43 -2 3 -44  -4 -3 12 10 0  0 3 0 36 1  3 1 2 3 4 |

**Скриншоты, демонстрирующие работу программы**

*Рисунок 2 – тестирование программы при исходных данных №1 (см. табл. 1)*

*Рисунок 3 – тестирование программы при исходных данных №2 (см. табл. 1)*



*Рисунок 4 – тестирование программы при исходных данных №3 (см. табл. 1)*

**Вывод:**

1) Разработан алгоритм для решения поставленной задачи. Составлена его схема в среде [draw.io](http://draw.io)(рис. 1)

2) Создан код программы по алгоритму в среде Vim

3) Проведено тестирование (см. табл. 1)

4) Тестирование показало корректность работы при заданных исходных данных (см. табл. 1 и рис. 2 - 4)