**Отчет по рубежному контролю №2**

*РК6-16Б*

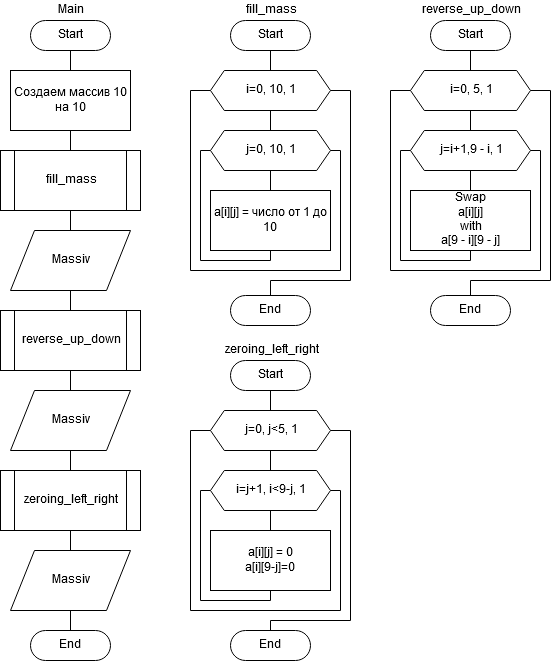
16. Петраков Станислав Альбертович

Преподаватель: Федорук Елена Владимировна

**Задача**

Разработать программу, которая запоминает в двумерном массиве, имеющем 10 строк и 10 столбцов, числа от 1 до 10, получаемые случайным образом. Поменять местами элементы массива, находящиеся в его верхней половине между главной и побочной диагоналями с симметричными элементами нижней половины. Обнулить элементы правой и левой частей массива, лежащие между главной и побочной диагоналями.

**Блок схема**



**Код программы**

//Boundary control No. 2

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

#define SIZEM 10

//main i=j

//secondary i+j=SIZEM-1

void fill\_mass(int a[][SIZEM])

{

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

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

a[i][j] = rand() % 10 + 1; // [1;10]

}

void output\_mass(int a[][SIZEM])

{

for (int i = 0; i < SIZEM; i++, printf("\n"))

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

printf("%2d ", a[i][j]);

printf("\n");

}

void reverse\_up\_down(int a[][SIZEM])

{

for (int i = 0; i < SIZEM / 2; i++) //Only upper part

for (int j = (i + 1); j < (SIZEM - (i + 1)); j++)//above the main, above the side

{

//swap

int temp = a[i][j];

a[i][j] = a[SIZEM - (i + 1)][SIZEM - (j + 1)];

a[SIZEM - (i + 1)][SIZEM - (j + 1)] = temp;

}

}

void zeroing\_left\_right(int a[][SIZEM])

{

for (int j = 0; j < SIZEM / 2; j++) //Only left part

for (int i = j + 1; i < (SIZEM - (j + 1)); i++)//Under the main, above the side

{

a[i][j] = 0;

a[i][SIZEM - j - 1] = 0;//mirror in the center

}

}

int main()

{

srand(time(NULL));

int table[SIZEM][SIZEM];

fill\_mass(table);

output\_mass(table);

reverse\_up\_down(table);

output\_mass(table);

zeroing\_left\_right(table);

output\_mass(table);

return 0;

}

**Тестовый запуск**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 1 | 2 | 5 | 7 | 5 | 2 | 5 | 9 | 7 |
| 1 | 6 | 4 | 6 | 5 | 6 | 7 | 3 | 8 | 7 |
| 3 | 10 | 10 | 8 | 8 | 7 | 10 | 10 | 8 | 7 |
| 5 | 9 | 4 | 4 | 8 | 9 | 4 | 7 | 6 | 9 |
| 10 | 2 | 5 | 9 | 5 | 5 | 9 | 7 | 10 | 10 |
| 10 | 10 | 1 | 9 | 7 | 8 | 4 | 1 | 10 | 6 |
| 2 | 8 | 1 | 7 | 6 | 9 | 5 | 3 | 2 | 10 |
| 10 | 3 | 6 | 7 | 10 | 1 | 7 | 3 | 3 | 1 |
| 1 | 3 | 7 | 9 | 9 | 8 | 3 | 9 | 4 | 4 |
| 1 | 10 | 2 | 6 | 6 | 1 | 1 | 4 | 3 | 9 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 4 | 1 | 1 | 6 | 6 | 2 | 10 | 7 |
| 1 | 6 | 9 | 3 | 8 | 9 | 9 | 7 | 8 | 7 |
| 3 | 10 | 10 | 7 | 1 | 10 | 7 | 10 | 8 | 7 |
| 5 | 9 | 4 | 4 | 9 | 6 | 4 | 7 | 6 | 9 |
| 10 | 2 | 5 | 9 | 5 | 5 | 9 | 7 | 10 | 10 |
| 10 | 10 | 1 | 9 | 7 | 8 | 4 | 1 | 10 | 6 |
| 2 | 8 | 1 | 7 | 9 | 8 | 5 | 3 | 2 | 10 |
| 10 | 3 | 6 | 10 | 7 | 8 | 8 | 3 | 3 | 1 |
| 1 | 3 | 3 | 7 | 6 | 5 | 6 | 4 | 4 | 4 |
| 1 | 9 | 5 | 2 | 5 | 7 | 5 | 2 | 1 | 9 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 3 | 4 | 1 | 1 | 6 | 6 | 2 | 10 | 7 |
| 0 | 6 | 9 | 3 | 8 | 9 | 9 | 7 | 8 | 0 |
| 0 | 0 | 10 | 7 | 1 | 10 | 7 | 10 | 0 | 0 |
| 0 | 0 | 0 | 4 | 9 | 6 | 4 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 5 | 5 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 7 | 8 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 7 | 9 | 8 | 5 | 0 | 0 | 0 |
| 0 | 0 | 6 | 10 | 7 | 8 | 8 | 3 | 0 | 0 |
| 0 | 3 | 3 | 7 | 6 | 5 | 6 | 4 | 4 | 0 |
| 1 | 9 | 5 | 2 | 5 | 7 | 5 | 2 | 1 | 9 |