|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** |
| Институт искусственного интеллекта |
| Кафедра Технологий Искусственного Интеллекта |

Практическая работа № 10

по дисциплине

«Процедурное программирование»

Обучающийся: Погосян С. А.

Группа: КВБО-07-23

Руководитель *Яковлев Д. А*

Москва 2023

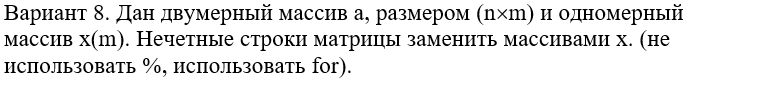
**Тема:** ОБРАБОТКА МАТРИЦ. ДИАГОНАЛИ И ТРЕУГОЛЬНАЯ ЧАСТЬ МАТРИЦЫ

**Цель практической работы** Разработка программ языке С++ с использованием двухмерных массивов, включая обработку главной и побочной диагонали, треугольную часть.

**Описание:**

1. **Теоретические сведения**
2. **Диагонали в двухмерных массивах**
3. **Главная диагональ**
4. **Побочная диагональ**

**Задание № 1:**

****

**Код программы:**

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <math.h>

#define N 10

#define M 10

int main() {

    srand(time(NULL));

    int \*\*ar;

    int \*x;

    ar = (int \*\*)malloc(sizeof(int) \* N);

    x = (int \*)malloc(sizeof(int) \* M);

    printf("X:\n ");

    for (int i = 0; i < M; ++i) {

        x[i] = rand() % (10 + 10 + 1) - 10;

        printf("%3d ", x[i]);

    }

    putchar('\n');

    printf("Array:\n");

    for (int i = 0; i < N; ++i) {

        ar[i] = (int \*)malloc(sizeof(int) \* M);

        for (int j = 0; j < M; ++j) {

            ar[i][j] = rand() % (100+100+1) - 100;

            printf("%5d ", ar[i][j]);

        }

        putchar('\n');

    }

    putchar('\n');

    printf("Result Array:\n ");

    for (int i = 0; i < N; ++i) {

        for (int j = 0; j < M; ++j) {

            if (i % 2 != 0) {

                printf("%5d ", x[j]);

            } else {

                printf("%5d ", ar[i][j]);

            }

        }

free(ar[i]);

        putchar('\n');

    }

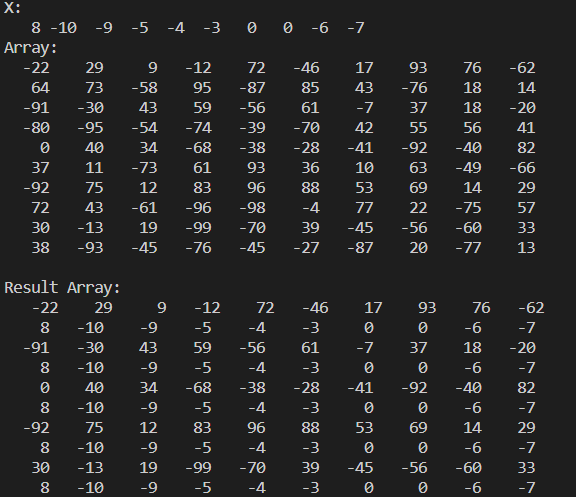
free(ar);

free(x);

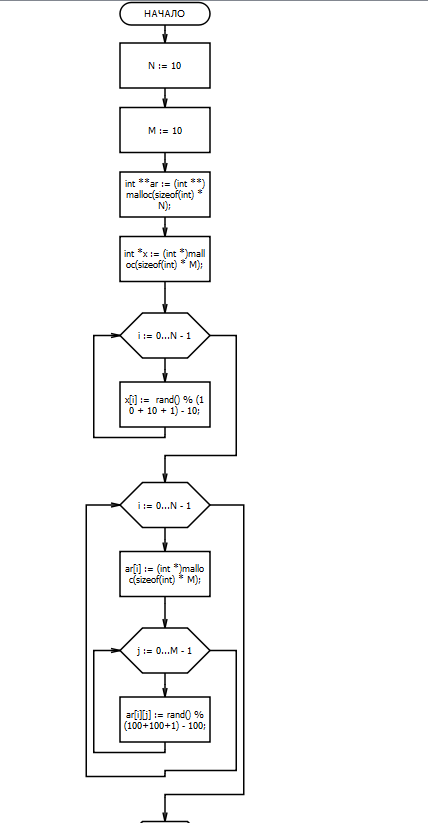
return 0;

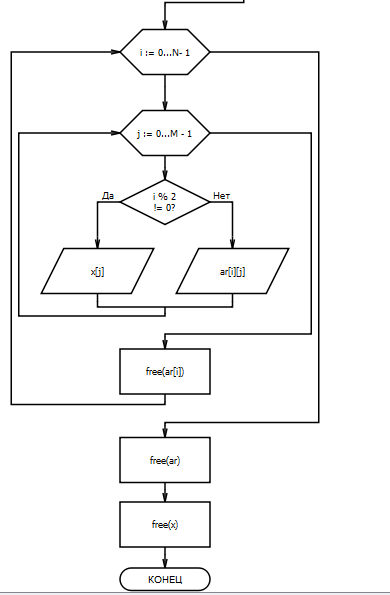
}

**Результат работы программы:**

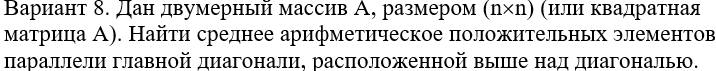


**Алгоритм в виде блок схемы:**

****

****

**Задание № 2:**



**Код программы:**

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#define N 2

int main() {

    srand(time(NULL));

    int \*\*ar;

    int total = 0, amount = 0;

    double average;

    ar = (int \*\*)malloc(sizeof(int) \* N);

    printf("Initial Array:\n");

    for (int i = 0; i < N; ++i) {

        ar[i] = (int \*)malloc(sizeof(int) \* N);

        for (int j = 0; j < N; ++j) {

            ar[i][j] = rand() % 10;

            printf("%3d ", ar[i][j]);

            if (i < j) {

                total += ar[i][j];

                amount += 1;

            }

        }

        putchar('\n');

    }

    putchar('\n');

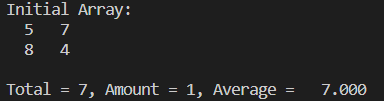
    average = (double)(total) / (double)(amount);

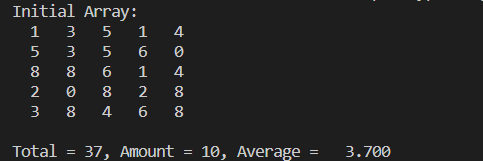
    printf("Total = %d, Amount = %d, Average = %7.3lf", total, amount, average);

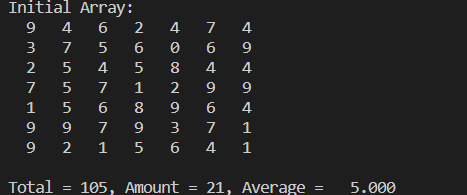
    free(ar);

}

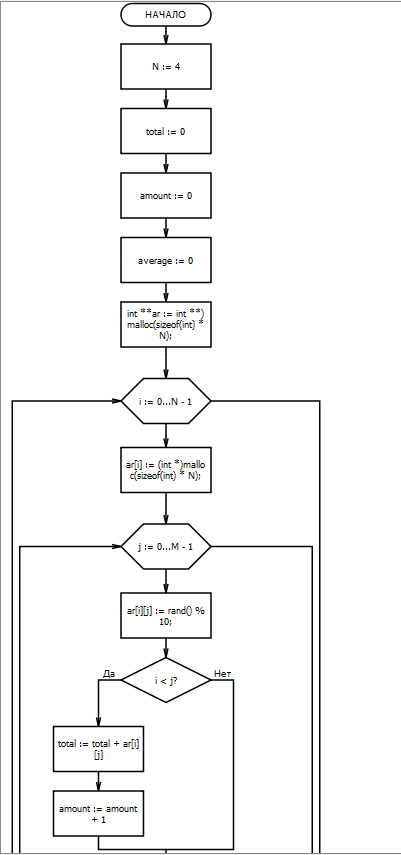
**Результат работы программы:**

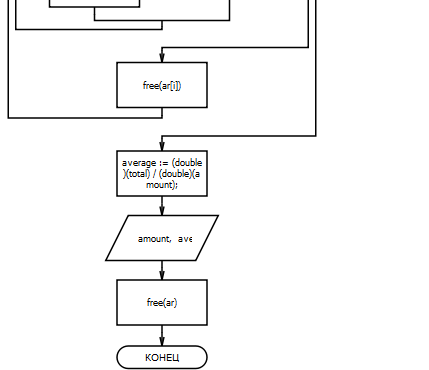




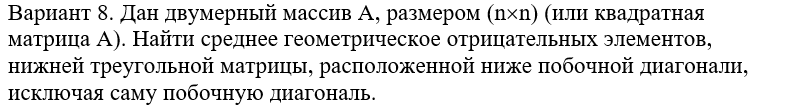


**Алгоритм в виде блок-схемы:**

****

****

**Задание № 3:**

****

**Код программы:**

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <math.h>

#define N 5

double get\_avg\_geometrical(int \*\*ar) {

    int total = 1;

    int amount = 0;

    for (int i = 0; i < N; ++i) {

        for (int j = 0; j < N; ++j) {

            if (i + j > N - 1 && ar[i][j] < 0) {

                printf("%8d [%d][%d]", ar[i][j], i, j);

                total \*= abs(ar[i][j]);

                amount += 1;

            }

        }

    }

    double result = (double)(pow(total, ((double)1 / (double)amount)));

    return result;

}

int main() {

    srand(time(NULL));

    int \*\*ar;

    ar = (int\*\*)malloc(sizeof(int) \* 5);

    printf("Initial Array:\n");

    for (int i = 0; i < N; ++i) {

        ar[i] = (int\*)malloc(sizeof(int) \* N);

        for (int j = 0; j < N; ++j) {

            if (i < j) {

                ar[i][j] = 0;

            } else {

                ar[i][j] = rand() % (10 + 10 + 1) - 10;

            }

            printf("%8d ", ar[i][j]);

        }

free(ar[i]);

        putchar('\n');

    }

    double avg\_geom = get\_avg\_geometrical(ar);

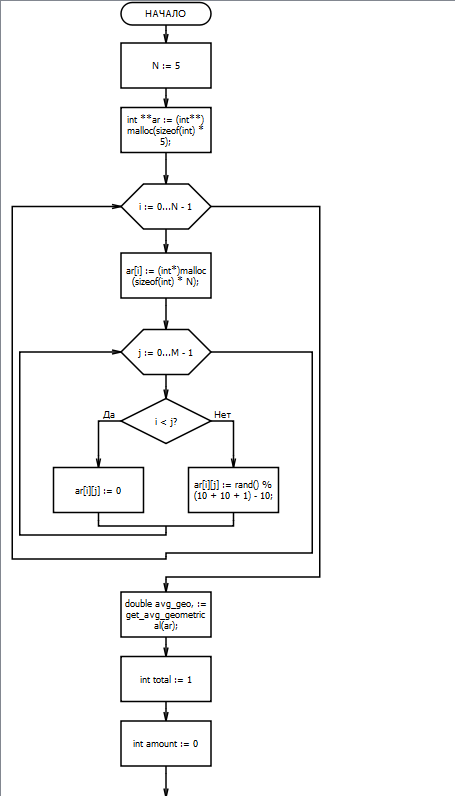
    putchar('\n');

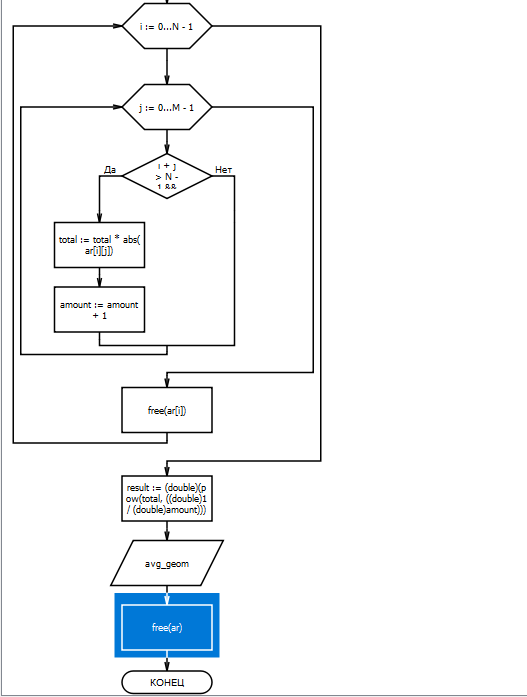
    printf("Average = %8.5lf", avg\_geom);

    free(ar);

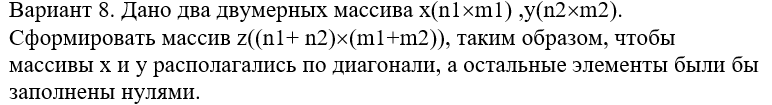
}

**Алгоритм в виде блок-схемы:**

****



**Задание № 4**

****

**Код программы:**

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#define N1 2

#define M1 2

#define N2 3

#define M2 3

int main() {

    srand(time(NULL));

    int \*\*x, \*\*y, \*\*z;

    x = (int \*\*)malloc(sizeof(int) \* N1);

    y = (int \*\*)malloc(sizeof(int) \* N2);

    z = (int \*\*)calloc(N1 + N2, sizeof(int));

    printf("X:\n");

    for (int i = 0; i < N1; ++i) {

        x[i] = (int\*)malloc(sizeof(int) \* M1);

        for (int j = 0; j < M1; ++j) {

            x[i][j] = rand() % (10 + 10 + 1) - 10;

            printf("%6d ", x[i][j]);

        }

        putchar('\n');

    }

    printf("\nY:\n");

    for (int i = 0; i < N2; ++i) {

        y[i] = (int \*)malloc(sizeof(int) \* M2);

        for (int j = 0; j < M2; ++j) {

            y[i][j] = rand() % (10 + 10 + 1) - 10;

            printf("%6d ", y[i][j]);

        }

        putchar('\n');

    }

    printf("\nZ:\n");

    for (int i = 0; i < (N1 + N2); ++i) {

        z[i] = (int \*)calloc(M1 + M2, sizeof(int));

        for (int j = 0; j < (M1 + M2); ++j) {

            printf("%d ", z[i][j]);

        }

        putchar('\n');

    }

    printf("\nZ Result:\n");

    for (int i = 0; i < N1; ++i) {

        for (int j = 0; j < M2; ++j) {

            z[i][j] = x[i][j];

        }

free(x[i]);

    }

    for(int i = N1 ;i < N1 + N2; ++i) {

        for(int j = M1; j < M1 + M2; ++j) {

            z[i][j] = y[i-N1][j-M1];

        }

free(y[i]);

    }

    for (int i = 0; i < (N1 + N2); ++i) {

        for (int j = 0; j < (M1 + M2); ++j) {

            if (z[i][j] != 0 && abs(z[i][j]) > 10) {

                z[i][j] = 0;

            }

            printf("%8d ", z[i][j]);

        }

free(z[i]);

        putchar('\n');

    }

    free(x);

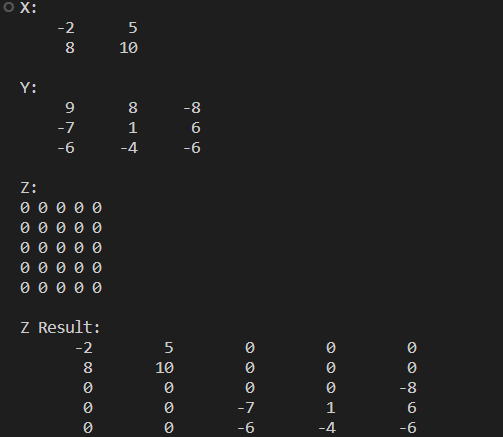
    free(y);

    free(z);

return 0;

}

**Результат работы программы:**



**Алгоритм в виде блок-схемы:**

