**Московский государственный технический**

**университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»

Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Основы информатики»

Отчет по лабораторной работе №8

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы ИУ5-13Б |  | преподаватель каф. ИУ5 |
| Лачина Екатерина |  | Аксенова М.В. |
| Подпись и дата: |  | Подпись и дата: |

Москва, 2023 г.

1. Задача

Создать квадратную матрицу A размером N\*N (где N вводится с клавиатуры), и заполнить её следующими значениями:

- все элементы главной диагонали равны 1;

- элементы, лежащие выше главной диагонали, вычисляются по формуле

A i,j = xi / (j!)i , а элементы, лежащие ниже главной диагонали, по формуле

A i,j = (-x)i / (j!)i, где i,j =1,2,…,N.

Для вычисления значений элементов матрицы использовать рекуррентные соотношения.

Реализовать алгоритм заполнения матрицы в виде функции.

В зависимости от размера матрицы и ширины поля вывода элемента матрицы, обеспечить удобное для пользователя отображение матрицы на экране. Оформить вывод матрицы размером N\*M на экран в виде функции с целью использования ее в последующих лабораторных работах для распечатки двумерных массивов.

Матрица должна передаваться в разрабатываемые функции через параметры.

Не изменяя кода функции вывода матрицы, распечатать матрицу в «научном» формате и в формате с фиксированной точкой с точностью 8 знаков после запятой.

Распечатать с помощью разработанной функции, используя вспомогательный массив указателей на строки, матрицу размером B[10][10], заданную с помощью оператора описания (нединамическую). Значение элементов матрицы В определяется соотношением: B[i][j]=i\*10+j.

Объясните, как передаются матрицы A и В в функцию вывода матриц на экран.

Вставьте в программу и объясните результаты выполнения следующих операторов

для матрицы В[10][10]:

cout<<B<<" "<<B[0]<<" "<<B[2]<<endl;

cout<<B[0][0]<<" "<<\*\*B <<" "<<\*B[0]<<endl;

cout<<\*(\*(B+1))<<" "<<\*B[1]<<endl;

cout<<\*(B[0]+1)<<" " <<\*(\*B+1)<<endl;

cout<<B[0][20]<<" "<<\*(B[0]+20)<<" "<<\*B[2]<<endl;

Прежде чем приступать к выполнению задания прочитайте приведенный ниже текст и разберите и выполните пример программы сортировки строк матрицы.

1. Разработка алгоритма

void fillMatrix(double\*\* matrix, int N, int M, int x) – функция заполнения матрицы по заданным формулам.

void printM(double\*\* matrix, int N, int M, int precision, char type) – функция вывода матрицы.

Matrix – матрица.

N – количество строк.

M – количество столбцов.

Precision – точность, количество знаков после запятой.

Type – тип вывода d/s (default/scientific).

Х – число, которое мы по формуле возводим в степень.

1. Листинг

#include <iostream>

#include <iomanip>

using namespace std;

void fillMatrix(double\*\* matrix, int N, int M, int x) {

matrix[0][0] = x;

for (int i = 1; i < M; i++) {

matrix[0][i] = matrix[0][i - 1] / (i + 1);

}

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

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

matrix[i][j] = matrix[0][j] \* matrix[i - 1][j];

}

}

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

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

if (i == j) {

matrix[i][j] = 1;

}

else if (i > j) {

matrix[i][j] = pow((-1), (i + 1)) \* matrix[i][j];

}

}

}

}

void printM(double\*\* matrix, int N, int M, int precision, char type) {

int screen\_size = 80;

int pole;

if (type == 'd')

pole = screen\_size / (precision + 5);

else

pole = screen\_size / (precision + 8 + 1);

int set = pole;

cout << "================================================================================" << endl;

while (pole <= M) {

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

for (int j = pole - set; j < pole; j++) {

if (type == 'd')

cout << fixed << showpos << setprecision(precision) << setw(precision + 5) << matrix[i][j];

else if (type == 's')

cout << scientific << showpos << setprecision(precision) << setw(precision + 8 + 1) << matrix[i][j];

}

cout << '\n';

}

pole += set;

cout << "\n";

cout << "================================================================================" << endl;

}

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

for (int j = pole - set; j < M; j++) {

if (type == 'd')

cout << fixed << showpos << setprecision(precision) << setw(precision + 5) << matrix[i][j];

else if (type == 's')

cout << scientific << showpos << setprecision(precision) << setw(precision + 8 + 1) << matrix[i][j];

}

cout << '\n';

}

}

int main(){

setlocale(LC\_ALL, "ru");

int N, M;

cout << "size N M: ";

cin >> N >> M;

char style;

cout << "Enter the d/s (default/scientific) - type of numbers: ";

cin >> style;

int precision;

cout << "numb after <,>: ";

cin >> precision;

double\*\* matrix = new double\* [N];

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

matrix[i] = new double[M];

}

int x = 1;

fillMatrix(matrix, N, M, x);

printM(matrix, N, M, precision, style);

double B[10][10];

for (int i = 0; i < 10; i++){

for (int j = 0; j < 10; j++){

B[i][j] = i \* 10 + j;

//b[i][j] = B[i][j];

}

}

cout << endl << endl << endl;

cout << "Matrix B:" << endl;

for (int i = 0; i < 10; i++) {

for (int j = 0; j < 10; j++) {

cout << fixed << setprecision(0) << B[i][j] << " ";

}

cout << endl;

}

cout << "Вывод результатов операторов для матрицы B[10][10]" << endl;

cout << B;

cout << " " << B[1] << " " << &B[3] << endl;

cout << \*(B[2] - 3) << " " << \*(B + 4) << " " << \*B[0] + 6 << endl;

cout << \*(\*(B + 6)) << " " << \*B[2] - 13 << endl;

cout << \*B[4] << " " << \*(\*B + 5) << endl;

cout << &B[4] << " " << &B << endl;

cout << &B[1][14] << " " << \*(B[6] + 17) << " " << \*(B[5] - 7) << endl << endl;

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

delete[] matrix[i];

}

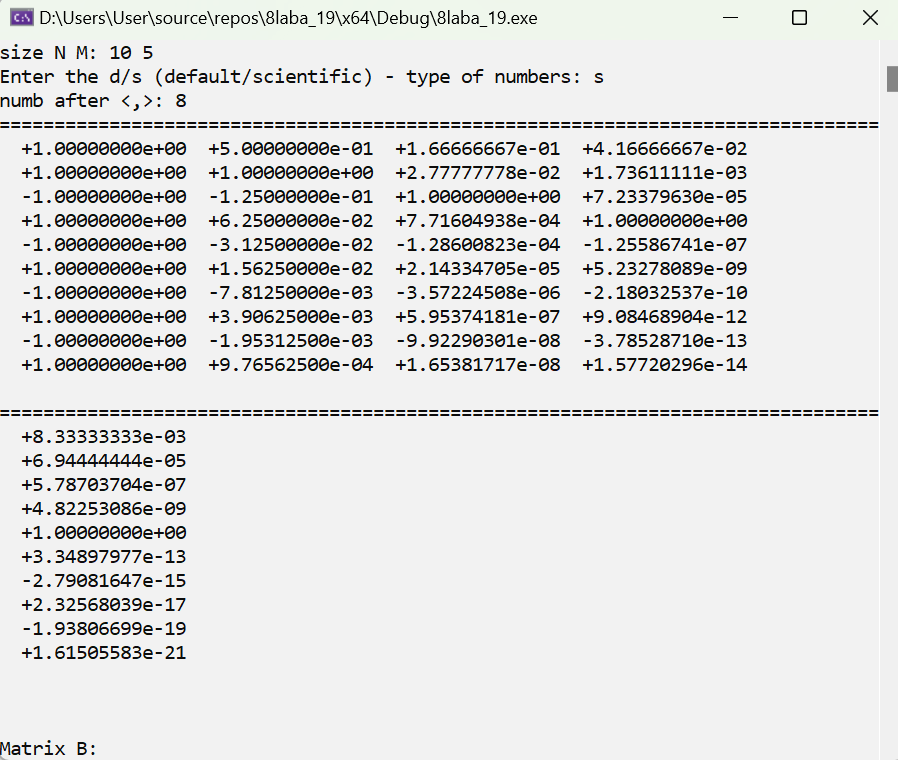
delete[] matrix;

system("pause");

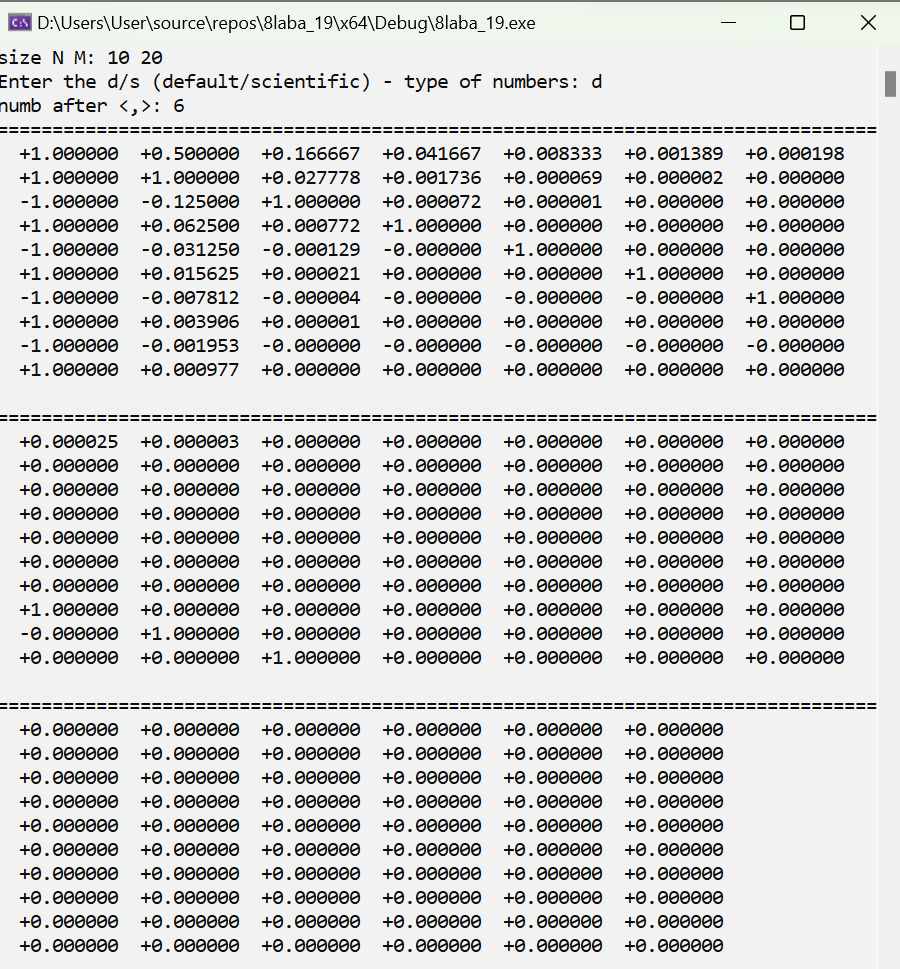
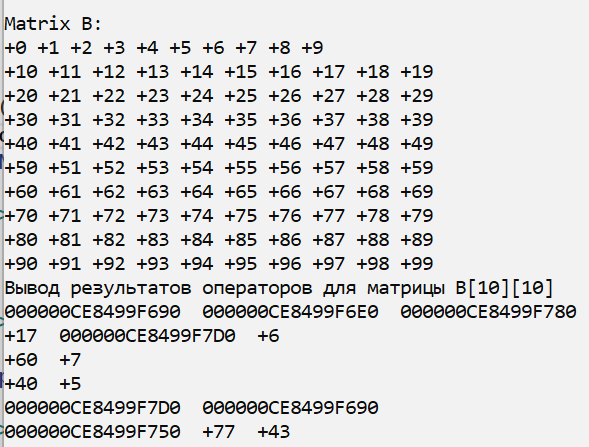
return 0;}

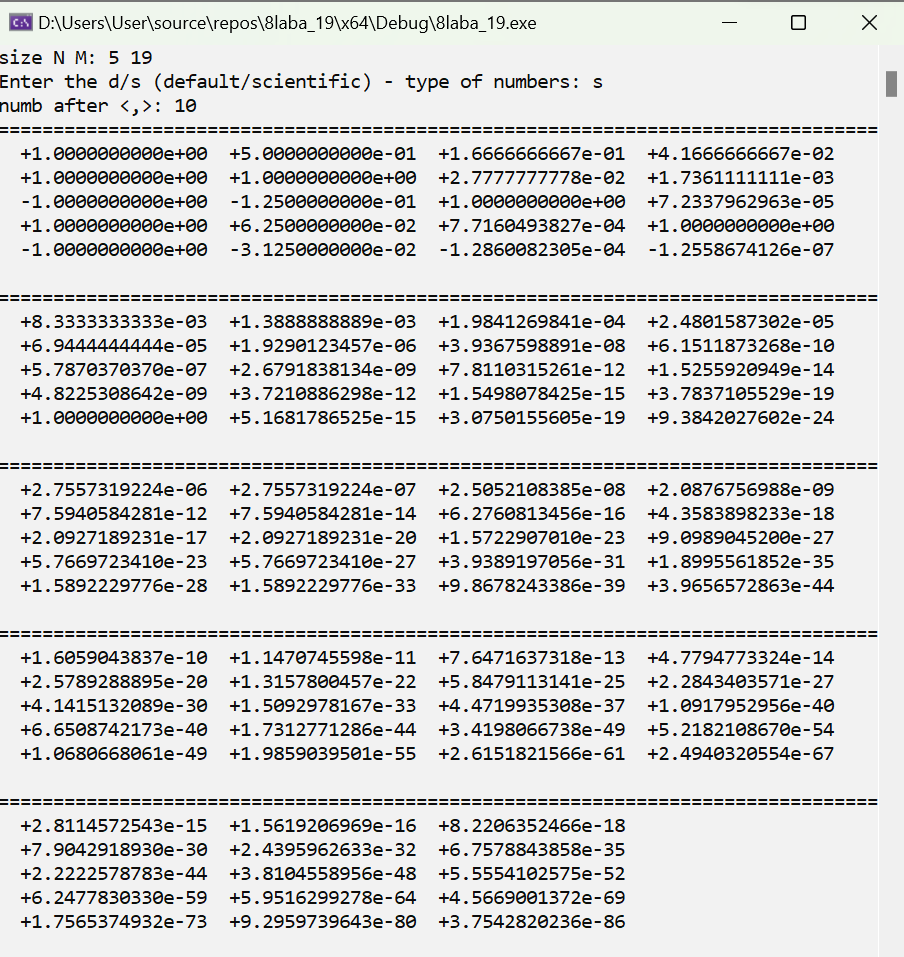
1. Тестирование

Пример 1



Пример 2



Пример 3