Московский государственный технический университет им. Н. Э. Баумана

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управление»

Курс «Основы программирования»

Отчет по лабораторной работе №7
«Обработка и печать числовой матрицы»

Выполнил:
Студент группы ИУ5-11Б
Алехин Сергей
Подпись и дата:

Проверил:
Преподаватель каф. ИУ5
Правдина Анна Дмитриевна
Подпись и дата:

Москва, 2018 г.

Задание

- создание двухмерных динамических массивов;
- обработка матриц;
- форматированный вывод матриц на экран;
-использование файлов для хранения матриц (на примере приведенной ниже программы сортировки строк матрицы);
- передача двумерных массивов в функцию через параметры;
- доступ к элементам матрицы через указатели и с помощью индексов;
- освоение технологии структурного программирования (разобрать и выполнить приведенную в описании работы программу сортировки строк матрицы).

Планируемое время выполнения работы - 6 часов.

Разработка алгоритма

Входные переменные:

1) int iNumColumns – количество столбцов в матрице;
2) int iNumLines – количество строк в матрице;
3) int iOutputOption – тип вывода на экран(fixed/scientific);
4) int iNumBeforeDot – количество знаков до запятой;
5) int iNumAfterDot – количество знаков после запятой;
6) double **dMatrix – матрица;
7) int x – переменная для нахождения элемента ij матрицы A(x = 1);
8) int fact – подсчет факториала;
9) const int iMatrixsize – размер матрицы B;
10) double B[][] – статическая матрица.

Функции:

1) void print – вывод матрицы на экран с возможностью выбора типа вывода;
    a. *Входные переменные:*
        i. double **dMatrix – матрица;
        ii. int iOutputOption – тип вывода на экран (fixed/scientific);
        iii. int iNumBeforeDot – количество знаков до запятой;
        iv. int iNumAfterDot – количество знаков после запятой;
        v. int iNumLines – количество строк;
        vi. int iNumColumns – количество столбцов.
    b. Локальные переменные:
        i. int iColumn – номер текущего столбца;
        ii. int iNumLength – длина слова (нужно для setw);
        iii. int iNumColumnsInRow – количество слолбцов в строке;

Текст программы

Файл Lab7.cpp

```
#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
using namespace std;
int main()
{
	setlocale(0, "RUSSIAN");
	srand(time(0));
	int iNumColumns, iNumLines, iOutputOption, iNumBeforeDot, iNumAfterDot;
	cout << "Количество строк = ";
	cin >> iNumLines;
	cout << "Количество столбцов = ";
```

```
	cin >> iNumColumns;
	cout << "Количество знаков после запятой = ";
	cin >> iNumAfterDot;
	cout << "Количество знаков до запятой = ";
	cin >> iNumBeforeDot;
	double **dMatrix = new double *[iNumLines];
	for (int i = 0; i < iNumLines; i++)
		dMatrix[i] = new double[iNumColumns];
	for (int i = 0; i < iNumLines; i++)
		for (int j = 0; j < iNumColumns; j++)
			dMatrix[i][j] = (double)rand() / RAND_MAX * (pow(10, iNumBeforeDot) -
0);
	cout << "Выберите вариант вывода матрицы: 1)fixed 2)scientific = ";
	cin >> iOutputOption;
	print(dMatrix, iOutputOption, iNumBeforeDot, iNumAfterDot, iNumLines, iNumColumns);
	for (int i = 0; i < iNumLines; i++)
		delete[] dMatrix[i];
	delete[] dMatrix;
	int x = 1, fact;
	cout << "Введите размер матрицы = ";
	cin >> iNumLines;
	cout << "Введите количество знаков после запятой = ";
	cin >> iNumAfterDot;
	dMatrix = new double *[iNumLines];
	for (int i = 0; i < iNumLines; i++)
		dMatrix[i] = new double[iNumLines];
	for (int i = 0; i < iNumLines; i++)
	{
		fact = 1;
		for (int j = 0; j < iNumLines; j++)
		{
			if (j != 0) fact *= j;
			if (i == j)
				dMatrix[i][j] = 1;
			else if (i < j)
				dMatrix[i][j] = pow(x, i) / pow(fact, i);
			else
				dMatrix[i][j] = pow(-x, i) / pow(fact, i);
		}
	}
	cout << "Матрица A" << endl;
	print(dMatrix, 1, 1, iNumAfterDot, iNumLines, iNumLines);


	const int iMatrixsize = 10;
	double B[10][10];
	for (int i = 0; i < iMatrixsize; i++)
		for (int j = 0; j < iMatrixsize; j++)
			B[i][j] = 10 * i + j;
	dMatrix = new double*[iMatrixsize];
	for (int i = 0; i < iMatrixsize; i++)
			dMatrix[i] = B[i];
	cout << "Матрица B" << endl;
	print(dMatrix, 1, 1, 0, iMatrixsize, iMatrixsize);
	cout << B;
	cout << " " << B[2]+4 << " " << B[1] << endl;
	cout << &B[3][5] << " " << **B+1 << " " << *B[3] << endl;
```

```
	cout << *(*(B + 2)) << " " << *B[4] << endl;
	cout << *(B[3] + 1) << " " << *(*B + 2) << endl;
	cout << B[0][20] << " " << *(B[0] + 13) << " " << *B[1] << endl;
	return 0;
}
```

Файл str.h

```
#pragma once
void print(double **, int, int, int, int, int);
```

Файл func.cpp

```
#include <iostream>
#include <iomanip>
#include <algorithm>

using namespace std;

void print(double **dMatrix, int iOutputOption, int iNumBeforeDot, int iNumAfterDot, int
iNumLines, int iNumColumns)
{
	int iColumn = 0, iNumLength, iNumColumnsInRow;
	if (iOutputOption == 1) iNumLength = iNumBeforeDot + iNumAfterDot + 2;
	else iNumLength = iNumAfterDot + 6;
	iNumColumnsInRow = floor(79 / (iNumLength));
	if (iNumColumnsInRow > iNumColumns) iNumColumnsInRow = iNumColumns;

	while (iColumn < iNumColumns)
	{
		for (int i = iColumn; i < min(iColumn + iNumColumnsInRow, iNumColumns); i++)
			cout << setw(iNumLength) << i + 1 << ' ';
		cout << endl;
		for (int i = 0; i < iNumLines; i++)
		{
			for (int j = iColumn; j < min(iNumColumns, iColumn +
iNumColumnsInRow); j++)
			{
				if (iOutputOption == 1) cout << fixed << setw(iNumLength) <<
setprecision(iNumAfterDot) << dMatrix[i][j] << ' ';
				else cout << scientific << setw(iNumLength) <<
setprecision(iNumAfterDot) << dMatrix[i][j] << ' ';
			}
			cout << endl;
		}
		iColumn += iNumColumnsInRow;
		cout << endl;
	}
}
```

Анализ результатов

№	Входные данные	Полученный результат

1	Вывод рандомной матрицы	Количество строк = 17 Количество столбцов = 18 Количество знаков после запятой = 2 Количество знаков до запятой = 2 Выберите вариант вывода матрицы: 1)fixed 2)scientific = 2 1 2 3 4 5 6 7 8 9 8.97e+00 7.91e+01 3.33e+01 8.79e-01 8.48e+01 9.30e+01 8.07e+01 5.37e+01 5.59e+01 9.14e+01 1.16e+01 9.93e+01 7.01e+00 7.28e+01 5.42e+01 8.73e+01 6.53e+01 2.22e+01 2.81e+01 7.45e+01 6.53e+01 2.04e+01 7.07e+01 2.97e+01 4.91e+00 2.63e+01 5.89e+01 3.37e+01 5.42e+01 1.56e+01 6.58e+01 5.30e+01 3.80e+00 8.67e+01 6.59e+01 9.48e+01 1.24e+01 5.61e+01 5.99e+01 9.17e+01 7.35e+01 2.63e+01 3.26e+01 9.20e+01 1.28e+01 5.73e+01 7.25e+01 3.76e+01 3.74e+01 2.13e+01 6.31e+01 7.55e+01 2.04e+01 4.17e+00 5.56e+01 7.20e+01 3.36e+01 3.23e+01 8.80e+01 7.47e+00 1.37e+01 7.85e+01 5.11e+01 8.35e+01 8.54e+01 8.87e+00 4.00e+01 2.52e+01 3.33e+01 6.99e+01 7.13e+01 8.30e-01 8.35e+01 3.49e+01 2.84e+01 2.58e+01 8.65e+01 3.75e+01 6.17e+01 7.75e+01 5.35e+01 8.26e+00 7.99e+01 9.43e+01 5.13e+01 6.59e+01 5.77e+01 4.84e+01 8.73e+00 8.70e+01 5.19e+01 4.01e+01 2.15e+01 3.63e+00 3.54e+01 8.32e+01 2.68e+01 3.62e+01 3.42e+01 3.46e+01 2.54e+01 7.02e-02 5.93e+01 7.64e+00 7.26e+00 5.64e+01 7.02e+01 7.96e+01 2.11e+01 5.42e+01 1.91e+01 7.60e+01 6.63e+01 1.44e+01 5.69e+01 8.81e+01 1.17e+01 1.90e+00 9.87e+01 6.11e+01 1.44e+01 8.57e+01 6.92e+01 4.81e+00 5.63e+01 2.55e+01 9.19e+00 6.81e+01 6.28e+01 2.67e+01 5.32e+01 8.33e+01 2.28e+01 8.89e+01 6.40e+01 3.95e+01 3.57e+01 4.75e+01 2.10e+01 9.19e+01 1.54e+00 6.71e+01 5.32e+01 9.81e+01 7.80e+01 5.22e+01 2.52e+01 3.77e+01 9.49e-01 7.20e+01 3.75e+01 7.22e+01 2.27e+01 10 11 12 13 14 15 16 17 18 4.82e+01 3.73e+01 3.26e+01 2.78e+01 4.17e+01 2.32e+01 9.38e+01 3.28e+01 2.44e+01 6.12e+01 1.86e+01 6.22e+01 1.49e+01 6.31e+01 8.03e+01 5.08e+01 6.65e-01 3.46e+01 7.17e+00 6.45e+01 1.47e+01 6.27e+01 7.71e+01 4.54e+01 9.57e+01 5.95e+01 3.39e+01 8.47e+00 1.60e+01 4.90e+01 2.94e+01 6.16e+01 5.91e+01 9.11e+01 5.12e+01 6.41e+01 1.20e+00 8.87e+01 2.11e+00 8.29e+01 8.16e+01 1.26e+00 2.75e+01 4.44e+01 3.44e+00 5.31e+01 8.91e+01 3.37e+01 9.04e+01 5.33e+01 7.08e-01 9.38e+01 2.82e+01 7.58e+01 3.01e+01 6.04e+01 4.42e+01 8.85e+01 5.85e+01 1.88e+01 4.08e+01 7.57e+01 8.88e+01 5.96e+01 4.58e+01 3.71e+01 5.15e+01 7.88e+01 2.99e+01 9.54e+01 3.93e+01 4.56e+01 3.95e+01 3.69e+01 3.55e+01 4.61e+01 1.02e+01 3.58e+01 3.67e+01 5.67e+01 1.09e+01 5.38e+01 6.15e+01 2.56e+01 8.70e+01 4.49e+01 8.53e+01 5.07e+01 5.65e+01 9.80e+01 7.00e+01 9.83e+01 8.18e+01 5.04e+01 6.42e+01 8.49e+01 2.33e+01 9.54e+01 9.88e+01 4.64e+01 7.00e+01 5.01e+01 3.72e+01 2.36e+00 5.09e+01 5.22e+01 2.15e+01 2.02e+00 5.92e+01 2.19e+00 4.74e+01 4.71e+01 1.74e+01 2.16e+01 4.38e+01 8.28e+01 6.78e+01 9.91e+01 2.77e+01 6.64e+01 9.39e+00 5.52e+01 8.77e+01 4.55e+01 6.01e+01 4.10e+01 8.85e+01 7.30e+00 9.70e+01 6.92e+01 4.10e+01 7.39e+01 4.07e+01 6.27e+01 2.20e+01 4.89e+01 3.49e+00 7.81e+01 1.18e+01 7.02e+01 7.03e+01 3.38e+01 3.32e+01 7.14e+01 1.40e+01 4.34e+01 9.90e+01 8.91e+01 3.47e+01 4.08e+01 7.16e+01 4.67e+01 2.37e+01
2	Вывод матрицы A	Введите размер матрицы = 13 Введите количество знаков после запятой = 3 Матрица A 1 2 3 4 5 6 7 8 9 10 11 12 13 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 1.000 -1.000 1.000 0.500 0.167 0.042 0.008 0.001 0.000 0.000 0.000 0.000 0.000 0.000 1.000 1.000 1.000 0.028 0.002 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 -1.000 -1.000 -0.125 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000 1.000 0.063 0.001 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 -1.000 -1.000 -0.031 -0.000 -0.000 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000 1.000 0.016 0.000 0.000 0.000 1.000 0.000 0.000 0.000 0.000 0.000 0.000 -1.000 -1.000 -0.008 -0.000 -0.000 -0.000 -0.000 1.000 0.000 0.000 0.000 0.000 0.000 1.000 1.000 0.004 0.000 0.000 0.000 0.000 0.000 1.000 0.000 0.000 0.000 0.000 -1.000 -1.000 -0.002 -0.000 -0.000 -0.000 -0.000 -0.000 -0.000 1.000 0.000 0.000 0.000 1.000 1.000 0.001 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000 0.000 0.000 -1.000 -1.000 -0.000 -0.000 -0.000 -0.000 -0.000 -0.000 -0.000 -0.000 -0.000 1.000 0.000 1.000 1.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 0.000 1.000
3	Вывод матрицы B	Матрица B 1 2 3 4 5 6 7 8 9 10 0 1 2 3 4 5 6 7 8 9 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 40 41 42 43 44 45 46 47 48 49 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 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
4	Вывод адресов и результатов	010FF440 010FF500 010FF490 010FF558 1 30 20 40 31 2 20 13 10

В этой работе мы научились выводить матрицу двумя способами, передавать статическую двумерную матрицу в функцию с помощью дополнительного массива, находить адреса и значения через указатели.