

Лабораторная работа №8

Ввод и вывод двумерных массивов

Оглавление

| | |
|------------------------------------------------|---|
| Цель | 1 |
| Инструкция: | 2 |
| Задания для самостоятельного выполнения: | 4 |
| Домашнее задание: | 4 |
| БЛОК А | 4 |
| БЛОК В | 5 |
| БЛОК С | 5 |
| Требования к оформлению программ: | 6 |
| Контрольные вопросы: | 6 |

Цель

Научиться объявлять динамические двумерные массивы, выделять память и освобождать память для них, заполнять и выводить их.

Инструкция:

Задача: считать матрицу размера n на m из файла и вывести на экран. Если файла нет или в нем недостаточно данных, или данные неподходящего типа – выводить на экран сообщение об ошибке.

Решение:

```
//-----main.cpp-----  
  
//Лабораторная работа №8: Ввод и вывод двумерных массивов  
  
#include <iostream>  
#include "matrio.h"  
  
#define n 5  
#define m 4  
  
using namespace std;  
  
int main()  
{  
  
    int **a=new int *[n];  
    for(int i=0;i<n;i++)  
        a[i]=new int [m];  
  
    InputFileMatr(a,n,m);  
    OutputFileMatr(a,n,m);  
  
    for(int i=0;i<n;i++)  
        delete[] a[i];  
    delete[] a;  
    a=NULL;  
  
    return 0;  
}  
  
//-----matrio.h-----  
  
#ifndef MATRIO_H_INCLUDED  
#define MATRIO_H_INCLUDED  
  
bool InputFileMatr(int** M,int n,int m);  
void OutputFileMatr(int** M,int n,int m);  
  
void InputDescMatr(int** M,int n,int m);  
void OutputDescMatr(int** M,int n,int m);  
  
void RandomMatr(int** M,int n,int m,int A,int B);  
  
#endif // MATRIO_H_INCLUDED
```

```

//-----matrix.cpp-----

#include<iostream>
#include<fstream>
#include<iomanip>
#include<stdlib.h>
#include<time.h>
#include "matrix.h"

using namespace std;

bool InputFileMatr(int** M,int n,int m)
{
    ifstream f("input.txt");

    if(!f)
    {
        cout<<"Error, there are no file to input matrix"<<endl;
        return false;
    }

    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            if(!(f>>M[i][j]))
            {
                cout<<"Error: not enough data or bad data in the file"<<endl;
                f.close();
                return false;
            }
    f.close();
    return true;
}

void OutputFileMatr(int** M,int n,int m)
{
    ofstream f("output.txt");

    for(int i=0;i<n;i++)
    {
        for(int j=0;j<m;j++)
            f<<M[i][j]<<" ";
        f<<endl;
    }
    f<<endl;
    f.close();
}

void InputDescMatr(int** M,int n,int m)
{
    cout<<"\n Please enter the matrix sized "<<n<<" x "<<m<<" :"<<endl;

    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            cin>>M[i][j];
    cout<<endl;
}

```

```

void OutputDescMatr(int** M,int n,int m)
{
    for(int i=0;i<n;i++)
    {
        for(int j=0;j<m;j++)
        {
            cout.setf(ios::fixed);
            cout.precision(2);
            cout<<setw(5)<<M[i][j];
        }
        cout<<endl;
    }
    cout<<endl;
}

void RandomMatr(int** M,int n,int m,int A,int B)
{
    srand(time(0));

    for(int i=0;i<n;i++)
        for(int j=0;j<m;j++)
            M[i][j]= rand()% (B-A+1)+A;
}

```

Задания для самостоятельного выполнения:

Набрать представленный в примере код программы, проанализировать результат и внести изменения:

1. Добавить функцию считывания матрицы с экрана, с прототипом:

```
void InputDescMatr (int** M, int n, int m);
```

И теперь, если входного файла нет или в нем недостаточно данных, или данные неподходящего типа – тогда считывать матрицу с экрана.

Домашнее задание:

БЛОК А

Задание на отметку «удовлетворительно»

Подготовить «заготовку» проекта для работы с двумерными массивами.

Внести в программу из лабораторной работы № 8 изменения:

1. Добавить функцию вывода матрицы в файл, с прототипом:

```
void OutputFileMatr(int** M, int n, int m);
```

- Добавить функцию заполнения матрицы случайными числами из интервала [A,B], с прототипом:

```
void RandomMatr(int** M,int n,int m,int A,int B);
```

Программа должна выполняться в следующей последовательности:

- Считывание с экрана размера матрицы n и m. Если пользователь вводит некорректные данные – требовать вводить их заново до тех пор, пока они не станут корректными. Некорректным в данном случае будет ввод отрицательного числа или нуля.
- Выделение памяти под двумерный динамический массив из n*m целых чисел.
- Вывод на экран меню вида:

```
-----Input Menu-----  
Screen input  
File input  
Random input
```

Если пользователь введет 1 – то считывать матрицу с экрана, если 2 – считывать из файла, если 3 – заполнять случайными числами из интервала [A , B], где А и В считывать с экрана.

- При считывании из файла, выполнять проверку его наличия. Если файла нет – то выводить сообщение об ошибке и просить пользователя ввести матрицу с экрана.
- Вывод на экран полученной матрицы.
- Вывод матрицы в файл.
- Освобождение памяти.

БЛОК В

Задания на отметку «хорошо»

Выполнить задание из **блока А** для двумерного массива вещественных чисел.

БЛОК С

Задания на отметку «отлично»

Выполнить задание из блока А, и внести изменения:

- Добавить функцию для выделения памяти под динамический двумерный массив.
- Добавить функцию для освобождения памяти.
- Использовать вывод матриц по формату – для вывода каждого элемента выделять по три символа.

Вызывать данные функции в программе.

Требования к оформлению программ:

1. **Содержание.** Программа должна делать то, что предусмотрено заданием. Не надо выполнять лишних действий, заданием не предусмотренных.
2. **Спецификация.** В преамбуле программы в комментариях указывать сведения:
 - Кто выполнил.
 - Что делает программа (кратко).
 - Что на входе (имена входных файлов и т.д.).
 - Что на выходе (что является результатом работы программы?).
3. **Ввод и вывод**
 - Приглашения к вводу (например, сколько чисел, какого типа и через какой разделитель нужно вводить).
 - Контрольный вывод (все введенные данные выводить на экран, и только после этого выполнять необходимые вычисления.)
 - «Защита от дурака». Проверять вводимые данные на корректность. Например, если необходимо считать количество чего – то, то эта величина не может быть отрицательной и т.д.
4. **Структура кода.** Набираемый код должен быть хорошо структурированным.
Использовать:
 - Отступы.
 - Комментарии – поясняют решение программы.
 - Осмысленные названия переменных.
 - Пояснения о назначении переменных в комментариях (кроме счетчиков).
5. **Декомпозиция кода**
 - Функциональная. Программу оформлять с помощью функций.
6. **Многофайловые проекты**
 - Все проекты должны состоять минимум из двух модулей: главного и подключаемого.

Главный модуль **main.cpp**. В нем оставить только функцию main, в которой вызывать функции, описанные в других модулях.

Модуль с описанием пользовательских функций **file.cpp** и **file.h**. В этот модуль перенести определение всех функций, необходимых для выполнения задания.

Контрольные вопросы:

1. Что такое двумерный массив?
2. Приведите пример объявления динамического двумерного массива из целых чисел. Выделите и освободите память.
3. Выведите значение элемента, находящегося в первом столбце и третьей строке матрицы.
4. Напишите функцию для заполнения матрицы единицами.
5. Напишите функцию вывода двумерного массива целых чисел на экран.
6. Напишите функцию ввода двумерного массива целых чисел с экрана.