Programming Design Homework Assignment S12-13

Due: 2022/6/4 23:00

※Notice:

1. Please follow the rules for homework assignments announced on the course website.
2. The standard template library and unrelated macros are not permitted in this assignment.

There are three questions in this assignment, question 1 is programming, and others are short answer question.

PART I Programming (80% in total)

- Programming formats：(16%)

  1. The structure of your program (4%)
  2. Clear and readable code layout (4%)
  3. Clear comments for understanding your program (4%)
  4. The copyright and short description of each question (4%)

- Question 1: (64%)

For this assignment, upgrade your structures in HWS10-11 into classes. The first class is also named MyMatrix which consist of a double pointer which point to a dynamic 2-dimensional array and the rows and columns of matrix as private members. The second class is MyValue which record the position of a value of matrix and which matrix it belonging to. The class declarations are shown below. Please copy the declarations into header file “MyMatrix.h” and “MyValue.h”, respectively, and finish them. Write the definitions of each member function in the file named “MyMatrix.cpp” and “MyValue.cpp”, respectively.

```
class MyMatrix
{
private:
    Float** matrix; //dynamic 2-dimensional array
    unsigned int rows;
    unsigned int columns;
    static int matrixCount; //count of objects of MyMatrix
public:
    MyMatirx(); //default constructor.
    MyMatrix(……); //two-arguments constructor, input the rows and columns to create
the dynamic array.
    MyMatrix(……); //copy constructor
    ~MyMatrix(); //destructor
```

```
        void output(……); //output the matrix orderly
        float get_value(……); //input the position, and get a value of an element
        …… kronecker_product(……); //input a matrix b to do the kronecker_product with
    "this" object and return the result
        MyValue get_minimum(……); //find the minimum element of the matrix, and
    encapsulate the information in an object of MyValue and return it back.
        static int getmatirxCount(); //output count of objects
    };
    class MyValue
    {
    private:
        MyMatrix* matrix; // the matrix that the value belong to
        unsigned int row; //the row of the value
        unsigned int column; //the column of the value
    public:
        MyValue(……); // multi-arguments constructor
        MyValue(……..); // copy constructor
        ~MyValue (); //destructor

        float the_value(); //use the row and column to output the value
        void output(……); //output information of "this" object
    };
```

Please create two matrices with class MyMatrix. The First matrix A is an m × n matrix and the second matrix B is a p × q matrix. The value of m, n, p and q are integer between 2 to 6 by random function. Use dynamic memory allocation to generate the space to store elements. The elements in the two matrices are also generated by random function (between -11.8 and +5.1). In main function, you shall use the above classes to create objects and call their member functions to calculate and output the following information on the console window.

1. The two original matrices.
2. The matrix after Kronecker product of two matrices (A⊗B).
3. Output the minimum element and its subscript of the above three matrices.
   The subscript is marked with 1. Ex: the minimum value is at the position of [2][2], it should be showed as (3,3).

The member function of each class could be void parameter or void function. DO NOT create any other member or member function in each class. Use "const" to protect reassignment of variables and parameters. Only the main function and function #6 and #7 can print the informations on the console window. Use MyMatrix::getmatirxCount() to check the amount of MyMatrix objects in your program before you releasing them.

PART II Short answer question (20% in total)

For short answer question, please answer the question in your words as detailed as possible. Submit as text file named by the question number, such as “HWS12-13-Q2.txt”.

- Question 2: (10%)

In question 1, which member functions of two classes you don’t need to define yourself? The compiler generated version of these functions are applicable. Why?

- Question 3: (10%)

In this semester, you wrote the assignment of matrix manipulations four times. In HWS7, the 2-dimesional array with dynamic memory allocation was first introduced. You wrote the all code in main function. In HWS8, you rewrote the requirements with several functions. In HWS10-11, the information of matrix and minimum value were encapsulated in two structure, respectively. Finally, in this assignment, you upgrade the structures into classes and create several member functions. Could you summarize the benefit of using class to create a matrix object? Please do not copy the features of object-oriented programming. You can compare these four assignments to find the idea.