



Engineering Applications General C++

Exercise

> 3. Class

1. Vector-Class

编写一个向量类:

类名: Vector;

数据成员:	int dim;	//向量的维数
	double * dataPtr;	//向量数据的指针
成员函数:	Vector();	//默认构造函数
	Vector(int N);	//构造函数
	Vector(const Vector & v);	//复制构造函数
	~Vector();	//析构函数
	void print(int w = 8);	//输出向量, setw(w)

> 3. Class

1. Vector-Class

编写一个向量类:

类名: Vector;

成员函数:
(续)

```
Vector & copy(Vector & v); //复制
Vector & copy(double x);   //所有元素用x赋值
double getData(int count); //读取元素
double setData(int count, double x); //元素赋值
int getSize();              //获取向量维数
Vector plus(Vector v);      //向量加法
Vector minus(Vector v);     //向量减法
double dotProduct(Vector v); //向量内积
Vector scale(double s);     //向量标量积(尺度变换)
```

> 3. Class

2. Matrix-Class

编写一个矩阵类：

类名： Matrix;

数据成员：	int _rows;	//矩阵的行数
	int _cols;	//矩阵的列数
	double **dataPtr;	//矩阵数据的指针
成员函数：	Matrix();	//默认构造函数
	Matrix(int row, int col);	//构造函数
	Matrix(const Matrix & m);	//复制构造函数
	~Matrix();	//析构函数
	void print(int w = 8);	//输出矩阵, setw(w)

> 3. Class

2. Matrix-Class

编写一个矩阵类：

类名：Matrix;

成员函数：
(续)

```
Matrix & copy(Matrix & m);    //复制
Matrix & copy(double x);    //所有元素用x赋值
double getData(int m, int n);    //读取元素
double setData(int m, int n, double x); //元素赋值
int getRows();                //获取行数
int getCols();                //获取列数
Matrix plus(Matrix & m);    //矩阵加法
Matrix minus(Matrix & m);    //矩阵减法
Vector product(Vector v);    //矩阵与向量相乘,
Matrix product(Matrix m);    //矩阵与矩阵相乘,
                             注意两种乘法相乘的前提
```

> 3. Class

3. Vector & Matrix Test

在主程序中进行以下测试：

```
int main()
{
    cout<<"Matrix A = B:"<<endl;
    Matrix A(3,5);
    A.copy(5.);
    Matrix B (A);
    B.print();

    cout<<"Vector v1:"<<endl;
    Vector v1(5);
    v1.copy(2.);
    v1.print();

    cout<<"Replace Matrix B at (2,2)
with 2:"<<endl;
    B.setData(2,2,2);
    B.print();

    cout<<"Vector v2 = B * v1:"<<endl;
    Vector v2(B.product(v1));
    v2.print();

    cout<<"A.Transpose:"<<endl;
    A.transpose().print();

    cout<<"C = B * A:"<<endl;
    Matrix C(B.product(A));
    C.print();
    return 0;
}
```