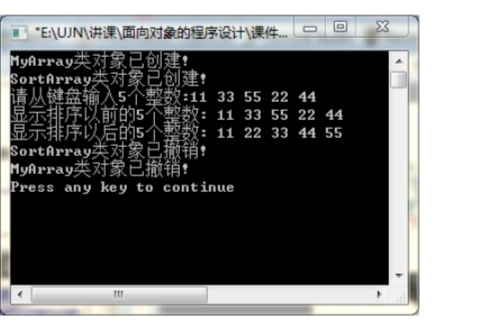
代码

#include  
#include  
using namespace std;  
class MyArray {  
public:  
MyArray(int length);  
~MyArray();  
void Input();  
void Display(string);  
protected:  
int \*alist;  
int length;  
};  
MyArray::MyArray(int leng)  
{  
if (leng <= 0)  
{  
cout << "error length";  
exit(1);  
}  
length = leng;  
alist = new int[length];  
if (alist == NULL)  
{  
cout << "assign failure";  
exit(1);  
}  
cout << "MyArray类对象已创建!" << endl;  
}  
MyArray::~MyArray()  
{  
delete[] alist;  
cout << "MyArray类对象已撤销!" << endl;  
}  
void MyArray::Display(string str)  
{  
int i;  
int \*p=alist;  
cout<<str<<length<<"个整数: ";  
for(i=0;i<length;i++,p++)  
cout<<\*p<<"";  
cout<<endl;  
}  
void MyArray::Input()  
{  
cout<<"请从键盘输入"<<length<<"个整数:";  
int i;  
int \*p=alist;  
for(i=0;i<length;i++,p++)  
cin>>\*p;  
}  
class SortArray:public MyArray  
{  
public:  
SortArray(int leng);  
~SortArray();  
void paixu();  
};  
  
SortArray::SortArray(int leng)  
{  
if (leng <= 0)  
{  
cout << "error length";  
exit(1);  
}  
length = leng;  
alist = new int[length];  
if (alist == NULL)  
{  
cout << "assign failure";  
exit(1);  
}  
cout << "SortArray类对象已创建!" << endl;  
}  
  
SortArray::~SortArray()  
{  
delete[] alist;  
cout << "SortArray类对象已撤销!" << endl;  
}  
  
void SortArray::paixu()  
{  
int i, t, j;  
int \*p = alist;  
for (i = 0; i < length - 1; i++)  
{  
for (j = i + 1; j < length; j++)  
{  
if (alist[i] > alist[j])  
{  
t = p[i];  
p[i] = p[j];  
p[j] = t;  
}  
}  
}  
for (i = 0; i < length; i++)  
cout << p[i] << " ";  
}  
int main()  
{  
MyArray a(5);  
a.Input();  
a.Display("显示已经输入的");  
return 0;  
}

结果

  
感想  
派生类的构造函数只负责对派生类新增的成员进行初始化，对所有从基类继承下来成员，其初始化工作还是由基类的构造函数完成。同样，对派生类对象的扫尾、清理工作也需要加入新的析构函数。  
构造函数。构造派生类的对象时，就要对基类数据成员、新增数据成员和成员对象的数据成员进行初始化。派生类的构造函数需要以合适的初值作为参数，其中一些参数要用于对派生类新增的成员进行初始化，另一些参数要分别传递给基类的构造函数和对象成员的构造函数。用于初始化相应的成员。  
析构函数。派生类的析构函数的功能是在该类对象消亡之前进行一些必要的清理工作。它没有类型也没有参数。  
通过上机操作，进一步理解体会派生与继承下的构造函数与析构函数的含义，实验中存在知识储备不完善的情况，但在同学的帮助下并结合教学课件，及时的解决了问题。