**程序代码**

#include<iostream>#include<string>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); } alist = new int [length]; length = leng; 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: void Sort(); SortArray(int leng):MyArray(leng) { cout<<"SortArray类对象已创建!"<<endl; } ~SortArray();};SortArray::~SortArray(){ cout<<"SortArray类对象已撤销!"<<endl;}void SortArray::Sort(){ int i,j,temp; for(i=0;i<length-1;i++) for(j=0;j<length-i-1;j++) { if(alist[j]>alist[j+1]) { temp=alist[j]; alist[j]=alist[j+1]; alist[j+1]=temp; } }}int main(){ SortArray s(5); s.Input(); s.Display("显示排序以前的"); s.Sort(); s.Display("显示排序以后的"); return 0;}**程序结果**

SortArray类对象已创建!

请从键盘输入5个整数

2

5

4

1

3

显示排序以前的5个整数：2 5 4 1 3

显示排序以后的5个整数：1 2 3 4 5

SortArray类对象已撤销!

MyArray类对象已撤销!

**感想心得**

通过本次上机实验我深刻理解了继承和派生的编写思路和规则，对于它的格式也非常熟悉，可以较为轻松地理解程序的思路,也对继承和派生的构造函数、析构函数的执行顺序有了更加清晰的了解。