代码

#include  
using namespace std;  
  
class Score  
{  
public:  
Score(int stnumber = 2)  
{  
if (stnumber > 100)  
number = 100;  
else number = stnumber;  
}  
  
void input()  
{  
for (int i = 0; i < number; i++)  
{  
//输入姓名  
cout << "请输入学生姓名:" << endl;  
cin >> name[i];  
//输入学生成绩  
cout << "请输入科目A成绩：" << endl;  
cin >> grade[i][1];  
cout << "请输入科目B成绩：" << endl;  
cin >> grade[i][2];  
cout << "请输入科目C成绩：" << endl;  
cin >> grade[i][3];  
}  
}  
  
void showallstu()  
{  
for (int i = 0; i < number; i++)  
{  
cout << "姓名：" << name[i] << " 科目A成绩：" << grade[i][1] << " 科目B成绩：" << grade[i][2] << " 科目C成绩：" << grade[i][3] << endl;  
}  
}  
  
void showstuavg(int m)  
{  
cout << "姓名:" << name[m] << " 平均成绩：" << (grade[m][1] + grade[m][2] + grade[m][3]) / 3 << endl;  
}  
  
void showavg(int i)  
{  
double all = 0;  
switch (i)  
{  
case 1:  
for (int m = 0; m < number; m++)  
{  
all += grade[m][1];  
};  
cout << "课程名称：A平均成绩：" << all / number << endl;  
break;  
case 2:  
for (int m = 0; m < number; m++)  
{  
all += grade[m][2];  
};  
cout << "课程名称：B平均成绩：" << all / number << endl;  
break;  
case 3:  
for (int m = 0; m < number; m++)  
{  
all += grade[m][3];  
};  
cout << "课程名称：C平均成绩：" << all / number << endl;  
break;  
}  
}  
void rank(int m)  
{  
int temp;  
char tempname[100];  
switch (m)  
{  
case 1:cout << "课程名：A" << endl;  
break;  
case 2:cout << "课程名：B" << endl;  
break;  
case 3:cout << "课程名：C" << endl;  
break;  
}  
for (int i = 0; i < number - 1; i++)  
{  
if (grade[i][m] > grade[i + 1][m])  
{  
temp = grade[i + 1][m];  
grade[i + 1][m] = grade[i][m];  
grade[i][m] = grade[i + 1][m];  
tempname[100] = name[i + 1][100];  
name[i + 1][100] = name[i][100];  
name[i][100] = tempname[100];  
}  
}  
for (int i = number - 1; i > -1; i--)  
{  
cout << "姓名：" << name[i] << " 成绩：" << grade[i][m] << endl;  
}  
}  
  
private:  
int number;  
char name[100][100];  
double grade[100][3];  
};  
  
int main()  
{  
Score stu;  
stu.input();  
stu.showallstu();  
stu.showstuavg(1);  
stu.showavg(2);  
stu.rank(2);  
return 0;  
}

结果

感想  
各种方法比较下来，冒泡排序相对简单，易实现，适用于轻量级的数据排序。冒泡排序简单来说就是从最底的数据开始冒,如果比他上面的数据小，就交换上去，然后再用第二下的数据比较，如果他比他上面的小，则交换，把小的冒上去。直到此到第一位置， 得到的就是最轻的数据，这个过程就像是冒泡一样，下面的和上面的比较，小的冒上去，大的沉下来。  
实验中也出现了许多小问题，主要还是因为自己不够细心。比如忘记中英文字符切换，或是少打花括号，大小写等等对应不上等问题。但在运行的时候会出现一些错误，只要跟着提示一个一个的改下去，其实是很简单的。

