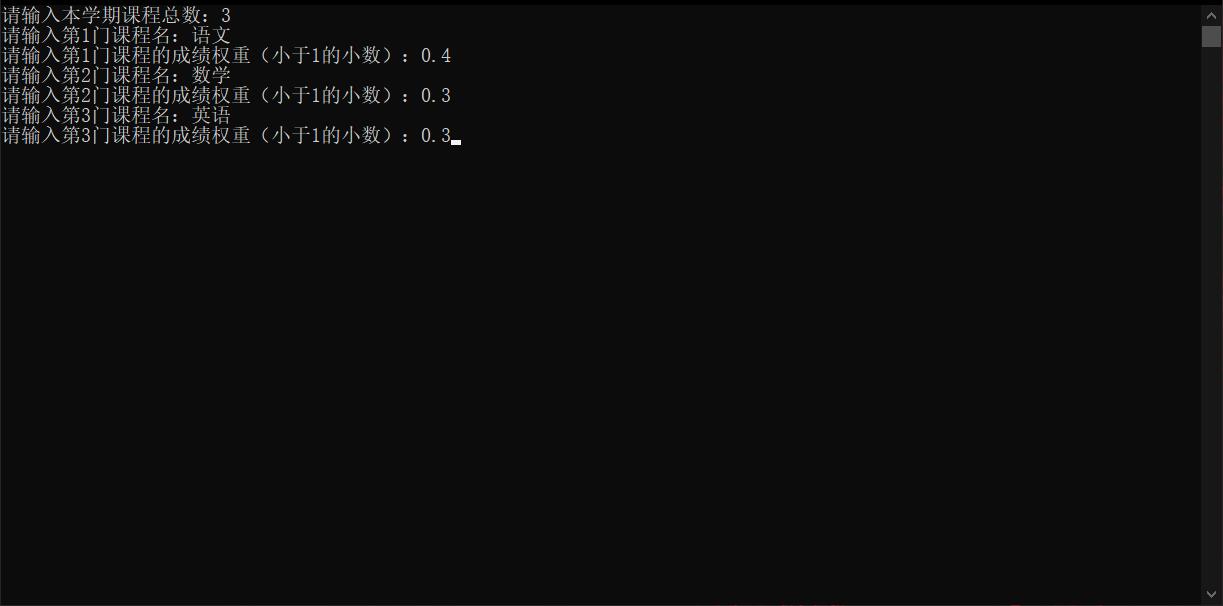
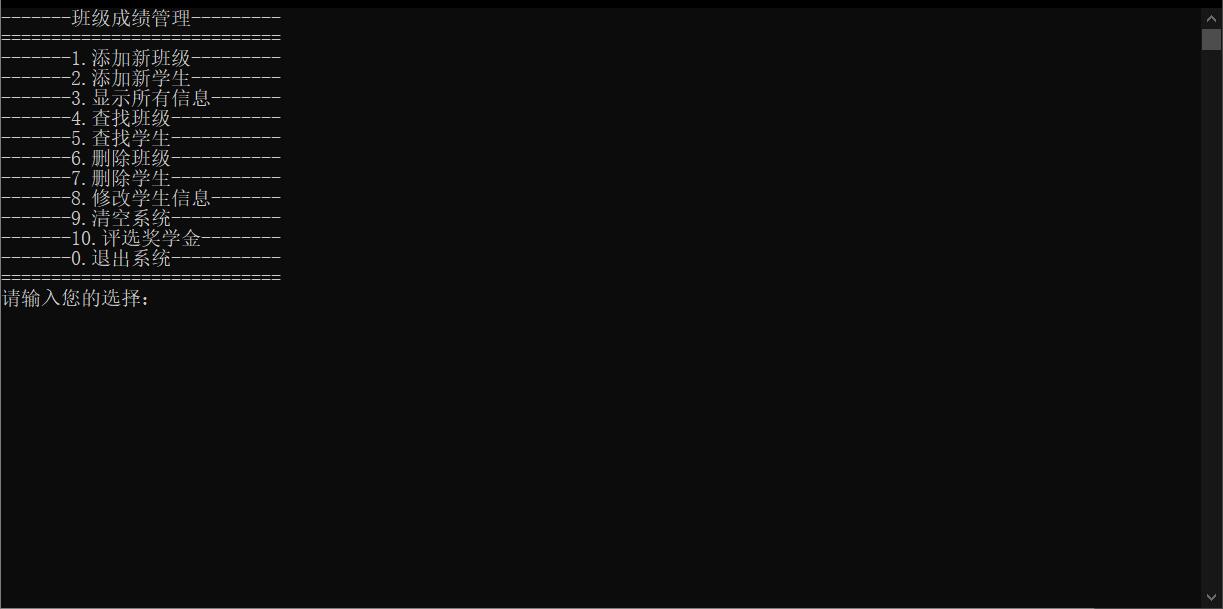
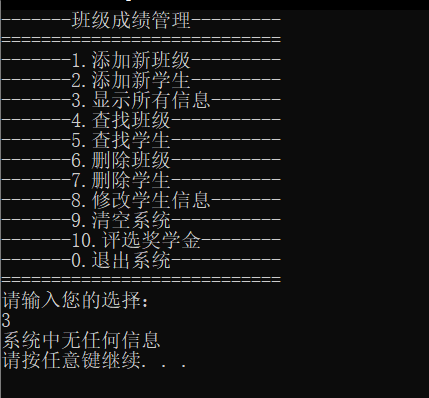
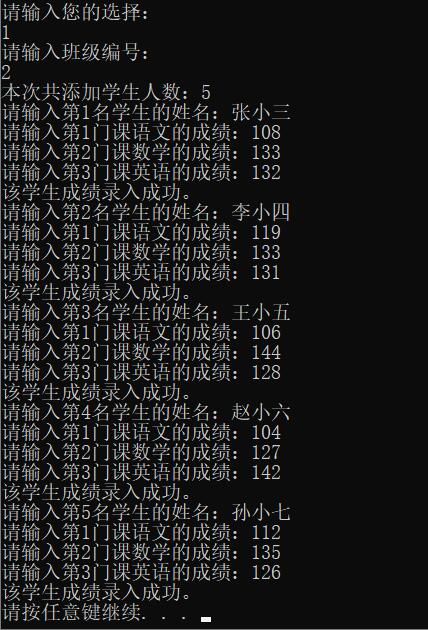
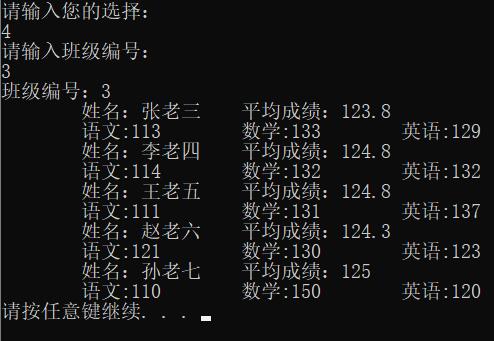
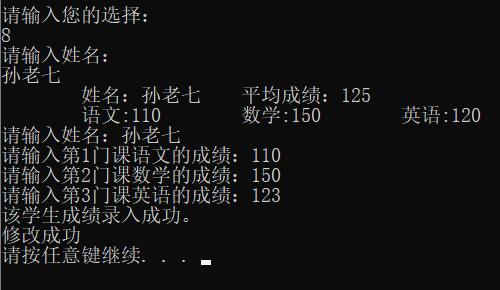
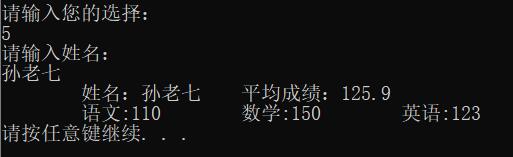
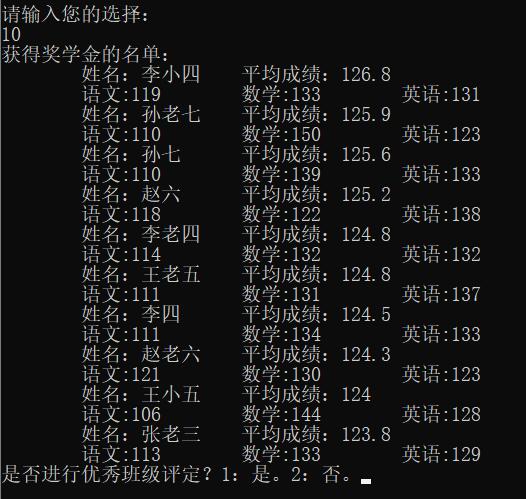
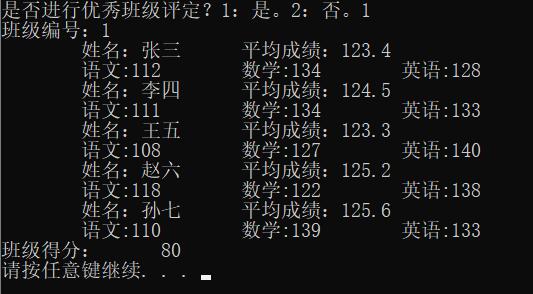
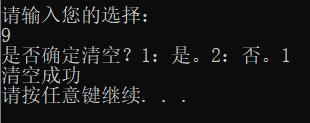
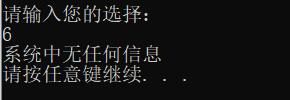
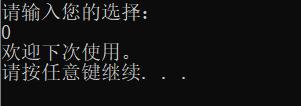
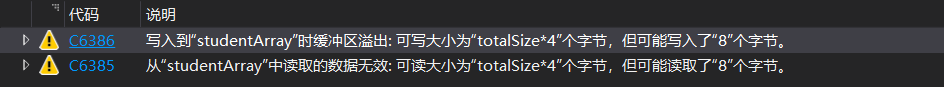
**“班级成绩管理系统”系统报告**

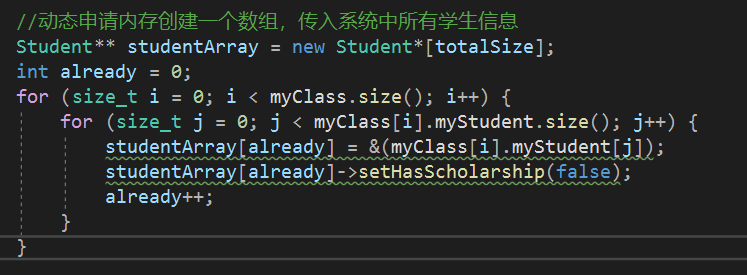
姓名：唐鹭

学号：2020302111391

1. **系统总体描述**

实现一个控制台应用用来进行班级成绩的管理，功能大致有：有班级、学生、成绩的增、删、改、查；进行学生排名，前十名获得奖学金；按照一定规则计算班级得分，获得最高分的班级评为优秀班级。

1. **相关功能描述**
2. 由用户设置所有课程的名称、成绩所占权重（一旦设定，则程序运行结束前不能更改）；
3. 显示菜单，根据用户输入判断执行什么操作；
4. 添加新班级（同时批量添加学生）；
5. 添加学生（先确定加入的班级）；
6. 删除班级；
7. 删除学生；
8. 查找班级；
9. 查找学生；
10. 修改学生信息；
11. 排名并评选奖学金；
12. 评选优秀班级（先执行功能10）。
13. **相关成果截图**
14. 初始化课程。
15. 显示菜单。
16. 未输入任何信息的时候，系统会打印提示信息。
17. 添加班级（以编号为2的班级为例）。
18. 查找班级（按照班级编号）。
19. 修改学生信息（按照学生姓名查找）。（删除功能类似，都是先查找，再执行操作）
20. 查找学生（按照学生姓名）。此时可以看到之前修改的学生信息已经成功得到修改。
21. 排名并评选奖学金。（此后会输出提示信息，询问用户是否需要继续进行评选优秀班级的操作。此时如果选择否，则会清屏并重新显示菜单。如此设计是因为，在计算班级得分的规则中需要知道班级中共有几人获得了奖学金，故评选优秀班级的操作必须在排名并评选了奖学金后进行。反之，则会出现默认为所有班级中获得奖学金的人数均为零的情况）
22. 评选优秀班级。
23. 清空系统与清空后的输出。
24. 退出系统。（退出系统后，在系统中保存的数据也丢失）
25. **作业成果总结与反思**
    1. 基本实现了要求的功能，进一步巩固了C++程序设计的相关基础知识。
    2. 为了用动态数组来实现储存课程、学生、班级的对象，学习使用了vector容器，以及它相关的函数如size()、clear()、push\_back()和erase()等。
    3. 为了实现根据成绩排名，自己动手实现了简单的冒泡排序算法。
    4. 根据现实世界三个类的特点，练习了对事物抽象的能力。设计了类似组合类的结构，例如Student类中含有一个Lessons类的vector动态数组作为其公有属性。
    5. 基本熟悉程序设计的流程，尤其是开始写代码之前的设计过程，开发过程中将类的定义的.h的头文件和类的实现以及程序主体main()函数分开在不同文件中，减小了单个文件的代码量，同时使开发过程更清晰。
    6. 有些设想的功能没有实现。比如课程一旦设置就无法修改，再如没有加入文件读写的功能，程序结束运行后所有数据就丢失了，这直接导致程序的实用性降低，而且增加了开发时调试的复杂程度，又如没有图形交互界面，等等。
    7. 程序中有部分遗留问题没有解决。除了在系统中没有信息时进行操作时程序会打印“系统中无任何信息”的提示信息外，本程序中绝大多数的输入没有检查输入合法性的检查。需要学习输入流的相关知识，了解cin的相关操作。
    8. Visual Studio 2019的编译器输出了如下警告：

在网上尝试搜索原因和解决方法后，我认为是在

这段代码中，为了实现给所有学生成绩排序，我动态申请了一片内存用来存放一个指针数组，其中的每个元素都指向系统中的某个学生。因为有班级和学生两个层次，所以我定义了一个局部变量already来存目前已被“传到”数组中的学生信息，而它的值同时也是下一个将要被“传入”的数组位置的下标，每“传入”一个学生，already都加1。编译器认为在此过程中存在already的值可能过大导致数组下标越界的危险，因此输出了这个警告。我仔细了检查程序，认为这个地方不会越界，但是不知道如何修改才能消除这个警告。