Project 1, Composition and Classes with Lists

Lukas Fuller

10/17/2017

Professor Andrew Mehri

**Abstract**

The purpose of project one is for the students to create a list of classes that will associate with students and their grade in each class. Through the completion of this project, we found how tedious it was to get and set each object to then call those objects in another file to finally print them. It forced me value the work required to code in this language especially when using pointers. This project was rather difficult for me to initially wrap my head around, but after coding it for a while it became clear to me what we had to do.

**Introduction**

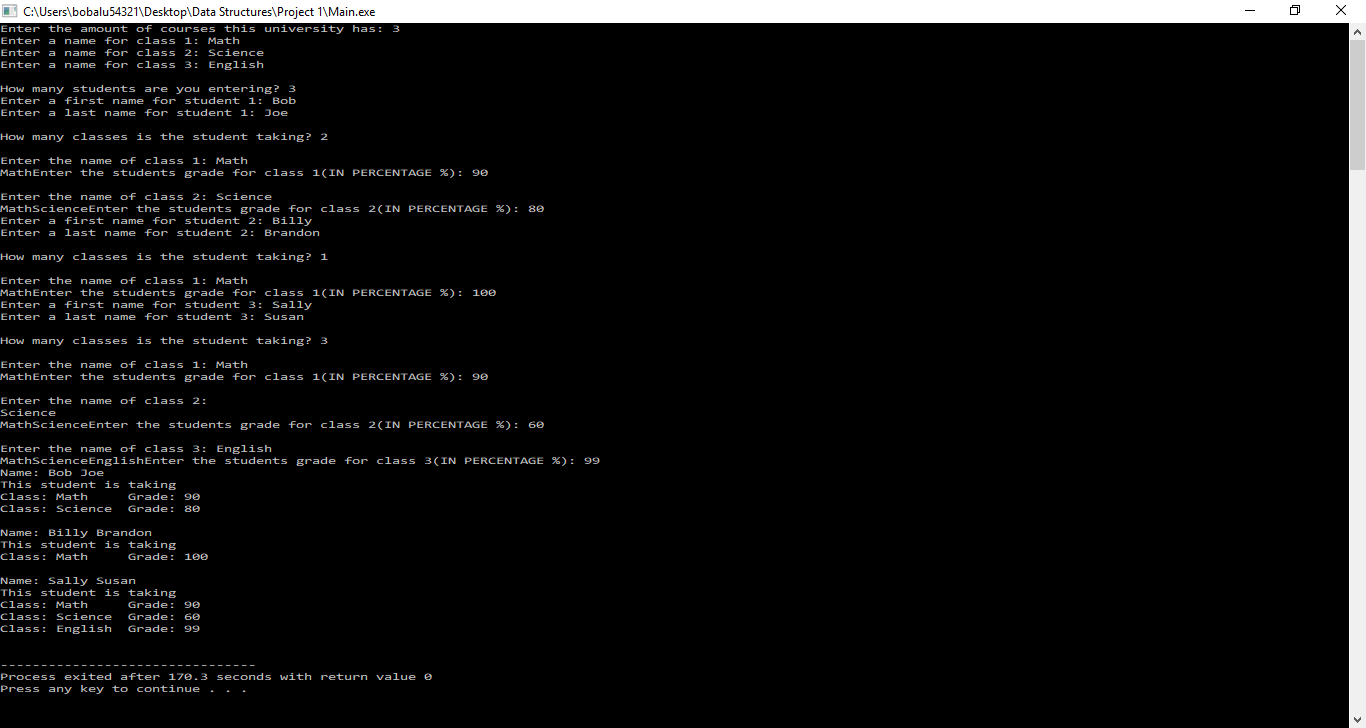
For project one, the main idea presented was: the program must have several instances of Student class, each student will hold a list of courses along with each grade, the program will ask the user to associate several courses to an existing student and grade, and finally print the result. These requirements made us choice to design our program by condensing each class and list functions into one file to save space. We also decided that each method would need to get and set each object in a separate file to then call these methods and display them together in another file.

**Procedure/Methodology**

Using methods to gather data from the user like the classes taken, number of students, and grades for each student are how we generated input from the user. The studentlist.cpp file handled the list itself, while creating instances of students as well as overloading a constructor with other important information, calling to the student.cpp for getters and setters for each variable, and printing the list of students. The student.cpp file handled each student’s first name, last name, creating each student, along with getters and setters for the information included in student.cpp, and displaying each student’s name along with the classes taken. The classlist.cpp file handled the creation of the class list itself and calls the classes.cpp file for getting and setting each variable. The classes.cpp file sets and gets each variable belonging to the class and prints the classes as well as the grade

**Results**

This is the output of the program when entering 3 classes Math, Science English, with 3 students Bob Joe, Billy Brandon, and Sally Susan. Bob has a 90 in Math and an 80 in Science, Billy has a 100 in Math, and Sally has a 90 in Math, 60 in Science, and a 99 in English:



In this example case it demonstrates the functionality of the program. It essentially runs through a set of for loops, do while loops, and if else statements to determine how many classes are offered, how many students, how many classes per student, each student’s grade for each class, and finally the print statement for the complete list of data.

**Analysis**

The print statements are all attributed to the getting and setting of each object in their respective classes.cpp and student.cpp. Through this, we are able to change each grade, student name, and class name in order to use them in the classlist.cpp and studentlist.cpp. These files handle the actual creation of each list object as well as the final print statements associated with each student and their classes’ grades. By calling each of these functions in the main file, we can declare each student’s name, the number of classes taken, the name of each class, and the grade for each class. There isn’t a huge structure to it, and it will only run once, which is a downside to the way we chose to finish it. However, the current code wouldn’t be too difficult to change with cases to make it look more user friendly. It would just require some redesign in the main function.

**Conclusion**

Through the completion of project one, we found the true difficulty in using lists. There are many lines of code required to operate some simple functions in another language. But C++ gives you much more control over the objects themselves, since you can directly use each object’s location in memory. Coding this project definitely helped my familiarity with the language, but it required a lot of effort from each of us to getting our files finished which made it very difficult at times. It was a pretty long project, but I can appreciate the effort required to make the final product.

**References**

Dale, N., Weems, C., & Richards, T. (2018). C plus data structures. Burlington, MA: Jones & Bartlett Learning.