MEC Part II

By:Cameron Brickett, Jay Gomes, Ishan Chadha

If our Grade calculator was to be made into a production level program we would have to change a couple of things around to make it work. The first thing that would have to be done would be a function that would ask the user how many different categories of grades there would be. For example one class would say quiz, test, homework, and Final. While another class could say test, midterm, final, and project. Each of these categories could be pushed onto a vector and then have another vector of the same length telling the program what the total amount of points the student could earn. Once we have all of the categories and total points saved in their own vectors, we would ask the user to input how much each individual quiz or exam would be worth. For example each quiz could be 20 points and the test could be out of 100 points. Again this would be put into its own vector. This would just be on the teacher side of the program so that they can get the correct grade scaling for the students.

On the student side of the program we would add graphics and create separate boxes so that it can be easily followed where the grades needed to go. Depending on what grade the students are at there could be different messages or graphics that appear on the screen when inputting grades or getting their final grade. When the program first starts we will have to add an instruction page for both the teachers and the students so that it can easily be worked with. Having a copy of the assignment available on the program for students and teachers to access would help in case of lost/damaged physical versions. A GPA calculator would be nice as well, and would require an additional algorithm to implement. Internet connectivity would help to keep everything updated online, and not require data to be stored locally. A queue would help in dealing with website traffic.

Lastly, we would like to add a fault checking system into our code. The inconvenience of restarting the program if a wrong input is entered could be tiresome. We can streamline this program to work better for students, especially if the data is large and will require input one at a time. If the wrong type or desired input is entered the program should account for that and display some error messages and prompt again for the input.