Matthew Power & Andrew Campbell

The algorithm used in our project was simple and straight to the point. However, it could be more efficient. There are multiple things that can be incorporated into our algorithm that we have learned from CSC 212 to better the algorithm—for example, using dynamic arrays to maximize compatibility. In our program, we only focused on helping a student analyze their grades/strengths/weaknesses after the semester was over. This meant all grades needed to be inputted from all sections (the final exam needed to be inputted or the output would not be correct). While this worked for the time being, if we were to scale this up, we would need to create a way to give the student the opportunity to not submit all of their grades, ie finding out what their grade is halfway through the semester. This would be through the use of dynamic arrays and altering the program. The program would have to recognize that no grade was inputted for a certain section and not contain that number in the final grade calculation. One way of doing this would be asking the student to enter -1 if they have not received a grade for that item yet.

Another aspect of the algorithm that could be improved would be implementing a way to upload the grades to the program either using the command line for instant feedback or creating a function to take in a bright space grade page and extract the numbers automatically. Another way of going about this is creating a .txt document as a template and having the student fill that out, making it really easy for students to check their grades quickly and over the semester the student can continually add their grades to check their progress at a moment's notice.