Seth Pitts
Ryan Demboski
10/4/19
Lab Report

1. **Problem Statement**
    a. The important features of the problem are:
        i. Having the input of a grade and then giving average grade, weighted grade, and final grade.
        ii. At the end, show the in order of all grades followed by final grade.
        iii. Also we added the main function to help organize the functions in a single one.
    b. The problem requirements
        i. Give the average grade
        ii. Give letter grade(when given the percent)
        iii. Give average weighted grade
        iv. Give final grade

2. **Planning**
    a. This problem was very troublesome in all categories. First the planning was to try and step by step all the def functions to allow for and easy flow between functions. First we started with our average function which we need to send all the scores together to find the sum then divide to find the average. For this we needed to use two different lists allowing for a correct list and total max list. Then we used calling of the lists to help find the average. Then we decided to work on the letter grade in which we used by having if statement coordinate with a certain percentage. Then one of the last definitions was the calculated average weighted. To do this we needed to get the weights of all assignments and allowed the averages to be multiplied to give us the final average with weights. Then last was to bring everything together and print out grades for homework, quizzes, tests, and then the final followed by the percent then after that would be their letter grade.

3. **Implementation and Testing**

Test of completed script running:

```
rad449@corellia:~/cs126/labs/lab4$ micro rad449_shp43_lab4_gradecalculator.py
rad449@corellia:~/cs126/labs/lab4$ python rad449_shp43_lab4_gradecalculator.py
Homework grade: 74 C
Quiz grade: 87 B
Test grade: 97 A
Final Score: 90 A
```

As shown in the picture above, the script outputs all of the required data with the correct information.

Proof of PEP8 Compliance:

```
rad449@corellia:~/cs126/labs/lab4$ pycodestyle-3 rad449_shp43_lab4_gradecalculator.py
rad449@corellia:~/cs126/labs/lab4$
```

4. **Reflection**
    a. As shown above with the example of the script running as well as the proof of it being PEP8 compliant, we believe the code was written in an effective manner. It did the job and delivered what we needed it to. However, it isn't the most efficient it could possibly be. We had some trouble understanding the concept of the main function, but once we understood it, we were able to correctly place our function calls and variables inside of the main function. Some things we wish that could have been done better would be writing more functions to cut back on rewriting the same code, and possibly find a better way to organize our data in our print statements. The letter_grade function could be written and called differently to make it run more effectively with other numbers. Rewriting the calls to the letter_grade function would make it a lot more effective if given different scores on the homework, quizzes, and tests. However, everything else was done efficiently and correctly and we believe our solution to finding the final score was a good way of solving it, by creating separate variables for each weighted category, and adding them all together.