Project 2 Report

1. The first difficulty I overcame was figuring out how to print my error codes after a hyphenated line, and not as a direct response to flawed input. I initially followed each line of my input with an “if” statement with an error output. This meant that an error output would be printed before the hyphenated line, with nothing coming after this line. I fixed this by instead combining all of my error code “if” statements into an “if ladder” that checks each input condition one after another and outputs an error code after the hyphenated line.

Another difficulty I faced was in declaring my variables. At first, I declared all of my variables as integers. I changed this after my program could not interpret multiple types of input. To fix it, I changed the types of variables I declared: I turned studentStatus and riderDestination into strings, and farePrice into a double. After this, I was able to use getline to correctly receive user input for these variables.  
  
A final challenge I faced was ensuring that my “if” statements were logical. I had to change the symbols within my conditions multiple times to account for the conditions necessary for certain discounts. For example, I needed to change riderAge > 18 to riderAge >= 18 for the student discount to be inclusive of the conditions within the project specs.

1. Test Data:

Full List:

**Age of rider: (**-5, 0, 10, 18, 20, 65, 70)

**Student? (y/n): (**n, y, N)

**Destination: (**Hollywood, \_\_ (blank))

**Number of zone boundaries crossed: (**-5, 0, 1, 2)

Reasons:

**Age of rider:** -5

^ This checks that my error code for “Student? (y/n)” is working correctly

**Age of rider:** 0

^ This makes sure that I am inclusive of 0 for fare prices

**Age of rider:** 10

^ This checks that my youth discount is working (if used in conjunction with zone boundaries 0 or 1)

**Age of rider:** 18, 20

^ These check that my student discount is working and inclusive of the borderline age of 18 (and that 18 isn’t mistakenly being used for the youth discount)

**Age of rider:** 65, 70

^ These check that my senior discount is working and inclusive of the borderline age of 65 (and that 65 is not being included within the student discount)

**Student? (y/n): y, n**

^ These check that the student discount is working correctly

**Student? (y/n): N**

^ This checks that the error code for studentStatus is working correctly

**Destination:**

^ This checks that my error code for Destination is working correctly

**Destination: Hollywood**

^ This can be a standard destination while testing other factors

**Number of zone boundaries crossed:** -5

^ This checks that my error code for zone boundaries is working correctly

**Number of zone boundaries crossed:** 0, 1, 2

^ These check that the three types of discounts for short trips are working, and that the regular price for longer trips is working correctly