1. Reflect: What are you proud of from this assignment? What was the most difficult part for you?

**Proud of:**

* Modular Design: Multiple functions such as, getValidMiles, calculateFare, printFare are used in this code which enhances the reusability, flexibility and readability of the code.
* Robust Input Handling: Multiple checks have been implemented in getValidMiles function so that it can handle multiple invalid input scenarios such as if user enters a value which is greater or less than maximum and minimum values, character.

**Most Difficult Part:**

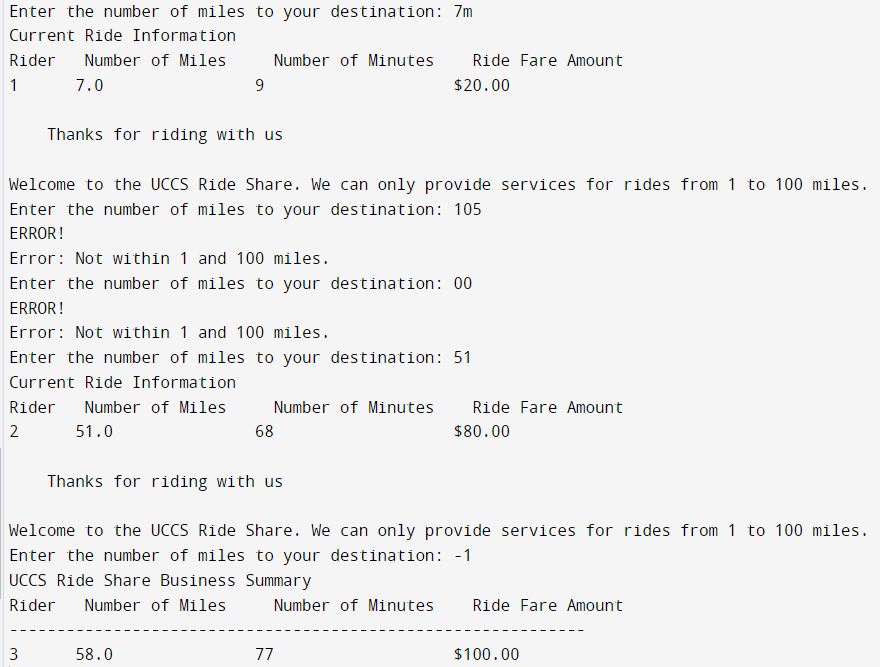
* Handling user input validation could be challenging, when considering different scenarios and edge cases.

3. Include related examples from either your design document or your code for each explanation. You can take screenshots or copy and paste your examples.

**3.1 Explain** functional completeness and what you did to ensure that in your solution.

The code handles invalid input scenarios, accepts valid input from user, calculate ride details, and at last provides a ride details summary which ensure that the program is aligned with the given acceptance criteria.

*Supported Screenshot:*



**3.2 Explain** functional correctness and what you did to ensure that in your solution.

In the calculateFare function we have implemented the base fare calculation as per given instruction. we are getting correct results regarding the ride fare as the expected and actual results are matching. The printFare function prints ride information with proper format.

*Supported Screenshot:*

A screenshot of a computer

Description automatically generated

**3.3 Explain** what you did to make your code maintainable and flexible.

The code is highly maintainable and flexible due to its modular structure. Each function has a specific responsibility which makes them easy to understand and modify. Constants are used in the code for easy computation.

*Supported Screenshot:*

A black text on a white background

Description automatically generated

**3.4 Explain** the secure coding standards you implemented in your solution.

While the code does not explicitly deal with sensitive data or security issues, but we have implemented the input validation logic so that it accepts only valid input to avoid unexpected behavior of the application.

*Supported Screenshot:*

A screenshot of a computer program

Description automatically generated

A screenshot of a computer program

Description automatically generated