Sadhana Vadrevu

CS 31

Professor Smallberg

Project 2 Report

The biggest obstacles I had to overcome in this project were mostly organizational ones. One major issue I ran into was how to check for error in the user input in an efficient manner. At first, I put all the checks at the end of the program in one big if-else statement, but this soon became very confusing and cumbersome. I kept losing track of where I was and what I was checking. That’s when I decided to have an if statement checking for an error each time the program read in an input from the user. This made my program much easier to read and manage. It was also hard for me to figure out an efficient way to calculate the rental charge. At first, my code was very repetitive and unproductive, as I did not use if-else statements to group similar situations that share rates together. However, once I drew out the pattern of rental charges on paper and was able to conceptualize it, it became a lot easier to put it into code.

**Test Cases:**

* Negative starting odometer reading (-100)
* Starting odometer reading is 0 (0, 500, 2, Fred, n, 8)
* Starting odometer reading is equal to ending reading (600, 600, 1, Lily, y, 6)
* Ending odometer reading is smaller than starting reading (600, 386)
* Negative number of rental days (500, 700, -6)
* 0 as number of rental days (500, 700, 0)
* Empty string as costumer name (680, 900, 3, )
* Costumer name is all spaces (400, 600, 8, , n, 7)
* Luxury status is not y or n (700, 900, 4, John, yes)
* Month is not an integer (500, 800, 6, Sally, y, 7.8)
* Month is not between 1 and 12 inclusive (500, 800, 3, Kim, n, 66)
* Month is 1 (600, 890, 5, Jack, n, 1)
* Month is 12 (680, 900, 2, Brad, y, 12)
* All input is valid (900, 1500, 6, Randy, n, 4)