605.201 Mini-Project 1:

Tortoise vs Hare Race Simulator

Noboru Hayashi

Johns Hopkins University

EN.605.201.85.SU20 Intro to Programming Using Java

June 28, 2020

Design And Analysis

1. General Program Design

This Java programming project requires the program to simulate many rounds of random steps that two contenders, a tortoise and a hare, move while they are racing. Also the standard output of the positions after each move and the final winner are required as well.

To achieve this functionality, I created two methods, each defining the series of operations which the main method runs multiple times: moveOnce() and renderPosition().

moveOnce() is a method taking a boolean argument: isTortoise and generating a random integer number between 1 and 10, then using it to simulate the how many positions that the contender moves next.

renderPosistion() takes positions of both a tortoise and a hare, also the length of the race, then uses a for loop to std out a string representing the positions of two contenders: ‘T’ for tortoise, ‘H’ for hare, or ‘OUCH!!’ if the two at the same position.

In the main method, a while loop runs until one of two contender’s positions reaches or passes the length of the race, MAX_LIMIT. Inside the loop, moveOnce is called, and the result of the simulation is added to contender’s positions. Then passing new positions to renderPosition() to render how the race is going. After exit the loop block, the main also examines which contender’s position reached the max, and prints out the winner, or possibly a tie.

Data structure used in the program is mainly integers, which are used to represent the position as an index. These indices are also compared with an iterative index of a for loop that runs inside renderPosition().

2. Alternative Approaches

I think the renderPosition() method can be put inside the main method since this method is just being called once in the while loop. However, separating some lines from the main, and moving to a helper method would help the programmer to read, understand and debug the codes if needed. Also I believe it makes the code much easier to read. So, I rather created the renderPosition method to separate the functionality.

3. Learning From This Project

From this project, I have learned how to divide the complex problem into many sub-problems, and this separation can be achieved by defining methods which have specific functionality each. I think from this experience, I can analyze the main requirements of future projects, and can attack them effectively by splitting into parts by features. And I believe this development process can assist the developers to narrow down the scopes of the problems.