UBI 503 Data Structures and Algorithms

5th Homework
Assignment Date: 11.12.2020
Due Date: 18.12.2020

From: C How to Program (7th Edition), Paul Deitel and Harvey Deitel, Prentice Hall, 2013.

7.14) (Slightly modified) ***(Simulation: The Tortoise and the Hare)*** In this problem you will recreate one of the truly great moments in history, namely the classic race of the tortoise and the hare. You will use random number generation to develop a simulation of this memorable event.

Our contenders begin the race at ``square 1" of 70 squares. Each square represents a possible position along the race course. The finish line is at square 70. The first contender to reach or pass square 70 is rewarded with a pail of fresh carrots and lettuce. The course weaves its way up the side of a slippery mountain, so occasionally the contenders lose ground.

There is a clock that ticks once per second. With each tick of the clock, your program should adjust the position of the animals according to the following rules:

Animal	Move Type	Percentage of the Time	Actual Move
Tortoise	Fast plod	50%	3 squares to the right
	Slip	20%	6 squares to the left
	Slow plod	30%	1 square to the right
Hare	Sleep	20%	No move at all
	Big hop	20%	9 squares to the right
	Big slip	10%	12 squares to the left
	Small hop	30%	1 square to the right
	Small slip	20%	2 squares to the left

Use variables to keep track of the positions of the animals (i.e., position numbers are 1-70). Start each animal at position 1 (i.e., the ``starting gate"). If an animal slips left before square 1, move the animal back to square 1. Generate the percentages in the preceding table by producing a random integer, i, in the range $1 \leq i \leq 10$. For the tortoise, perform a ``fast plod" when $1 \leq i \leq 5$, a ``slip" when $6 \leq i \leq 7$, or a ``slow plod" when $8 \leq i \leq 10$. Use a similar technique to move the hare.

Begin the race by printing
BANG !!!!!
AND THEY'RE OFF !!!!!

Then, for each tick of the clock (i.e., each repetition of a loop), print a 70-position line showing the letter T in the position of the tortoise and the letter H in the position of the hare. Occasionally, the contenders will land on the same square. In this case, the tortoise bites the hare and your program should print OUCH!!! beginning at that position. All print positions other than the T, the H, or the OUCH!!! (in case of a tie) should be blank.

After each line is printed, test if either animal has reached or passed square 70. If so, then print the winner and terminate the simulation. If the tortoise wins, print "TORTOISE WINS!!! YAY!!!". If the hare wins, print "Hare wins. Yuch". If both animals win on the same tick of the clock, you may want to favor the turtle (the ``underdog''), or you may want to print "It's a tie". If neither animal wins, perform the loop again to simulate the next tick of the clock.

In your program, you are expected to provide appropriate functions in which animal moves are simulated. Use pointers of the variables representing the animals within the functions.
(100 points)

Important Notes:
1. All source codes and related homework reports should be submitted via Ege Ders platform: Derslerim → Veri Yapıları ve Algoritmalar - 430882-4 → Hafta 6: Pointers → Homework 5. Please also send a copy of your files to sinem.getir@ege.edu.tr in the format of ogrenci_no_odev_no.zip
2. Do not forget to include appropriate comments in the source codes. Hence the grader can easily understand the program during his/her assessment.
3. Write the programs in a simple and straightforward manner by considering structured programming.
4. Each report should include the printout of the related source code and two or more screenshots (depending on the illustration requirements) which exemplify execution of the programs.