CS 3335 (The C Programming Language) [Fall 2019]

Coding Assignment 1

Due date: 9:00 p.m., Friday, September 6, 2019

Task:

Write a C-program to guess a random integer, between 0 and 100, that has been generated by the program. The program should restrict user for fixed number of trials and should indicate the user how many trial he/she has to guess the number. Each incorrect guess should be identified as "Too high" or "Too low". After each incorrect trial, the program should indicate the user how many trials are left. If the number is guessed correctly within the trial limit, display the correct number to confirm the correct guess and show appropriate success message with number of trials used before terminating the program. If number of attempts is exhausted and the number is not successfully guessed, the user should be notified about that (like "you have exhausted all your trials", or some similar message) and the correct number should be displayed before terminating the program.

Use global constant to set the maximum number of trials allowed. Set it to 5, or 7, or 10, but please don't set it to more than 10.

Note: *The C library uses a rand() function to generate a pseudo-random integer between 0 and RAND_MAX (where RAND-MAX is largest signed 32-bit integer). However, this will generate an integer from the same pseudo-random sequence every time. Therefore, to use rand() effectively (that is, to use a different pseudo-random generator each time), you need to use srand() function, which returns void. Therefore, just call* ***srand(time(NULL))*** *in main() once only, and then use rand() as many times as you want. Both functions are in the header file* ***stdlib.h*** *that you need to include using #include<> statement.*

Submission instruction: (must be followed if needs to be graded)

Name the files as ***a1lastname.c*** where *lastname* is your last name. At the top, write your name, course (CS3335), and semester-year (Fall 2019) as C-comments (use /*… */). Submit the file through BlazeVIEW submission box associated to the assignment.