

Programming Exercise 6.7

Over-Under Guessing Game, v.2.0

Purpose. Learn how to use loops.

Supplemental. Read about “randomizing” in www.rdb3.com/cpp/exercises/Gaming.supplemental.pdf.

Requirements. Rewrite your `overUnder1.cpp` from exercise 6.6 as `overUnder2.cpp`. Modify the program to allow the human player to guess again if their guess was not correct, and continue until it is correct.

Algorithm.

Call `srand`

Get and store the number to be guessed (1-100) using the random number generator

Output the computer's challenge to the human, to try to guess the randomly-selected number

Loop starts here

Input and store the human's guess as a whole number

If the human's guess is EQUAL TO the randomly-selected number
Output the randomly-selected number with a label
Break from the loop

If the human's guess is LESS than the randomly-selected number
Output something like "That's too low -- guess again"

If the human's guess is GREATER than the randomly-selected number
Output something like "That's too high -- guess again"

Loop back from here

Example. User input in blue:

```
I'm thinking of a number between 1 and 100. Guess what it is: 50
That's too high -- guess again: 25
That's too low -- guess again: 37
That's too high -- guess again: 31
That's too high -- guess again: 28
That's right -- it's 28
```