

Robert Navarro, Erin Potter, Joyce Chew
CS 161
Project Group 24
Assignment 3 (Group)

Understanding:

This program should be able to:

- Store a number that is entered by a user
- Prompt the player to guess the original number
- Evaluate whether or not the guess is higher or lower than the original number
- Instruct the player to guess higher or lower based on the relationship between the guess and the original number
- Continue to prompt the player to enter a number and evaluate its relationship to the original number until the player guesses the original number
- Count and display the number of guesses it takes the player to guess the original number

The use of relational operators, if/else statements, loops, and counters are all new techniques this week. The relational operators (< and >) are used to determine whether the player's guess is higher or lower than the original number. The if/else statements are used to instruct the player to guess a higher or lower value based on the relationship of the guess to the original number. A loop is used to keep asking the player to guess until the original number is guessed and a counter is used to track the number of guesses. The ability to nest if/else statements and loops is also a new technique for this week. This technique will be necessary to set up if/else conditions within the loop.

Testing Plan:

Description	Expected Outcome
Guess = Original on 1 st try	Should display "you guessed it in 1 tries."
Orig = -, Guess = +	Should display "lower," ask player to guess again.
Orig = +, Guess = -	Should display "higher," ask player to guess again.
Correct guess	Display Message w Correct # of guesses

Test	Result
Pick a positive number and have the first guess be a negative number.	The program should say that the number is "higher" and then guide the user until they guess the correct number.

Pick a positive number and have the first guess be higher than the than the picked number.	The program should say that the number is “lower” and should guide the user until they guess the correct number.
Pick a positive number and have the first guess be a positive number but lower than the picked number.	The program should say that the number is “higher” and then guide the user until the guess the correct number.
Pick a negative number and have the first guess be a negative number that is lower than the number that was picked.	The program should say that the number is “higher” and then guide the user until the guess the correct number.
Pick a negative number and have the first guess be a negative number that is higher than the number that was picked.	The program should say that the number is “lower” and should guide the user until they guess the correct number.
Pick a negative number and have the first guess be 0.	The program should say that the number is “lower” and should guide the user until they guess the correct number.
Pick a positive number and have the first guess be 0.	The program should say that the number is “higher” and then guide the user until the guess the correct number.

Design:

Set the counter to 1.

Get a number from the user.

Prompt the player to guess the original number.

While the guess is not equal to the original number:

Increment the counter +1.

If the guess is lower than the original number,
tell the player “higher.”
Prompt the player to guess again.

Else,
tell the player “lower.”
Prompt the player to guess again.

Tell the player “you guessed it in x tries” where x is the final number from the counter.