IT 140: INTRODUCTION TO SCRIPTING

Cathrine Karangi

Southern New Hampshire University

IT 140: Introduction To Scripting

Professor: Wayne Sevigny

May 28, 2025

**Higher/Lower Game Pseudocode**

BEGIN

// Step 1: Welcome the user

DISPLAY "Welcome to the higher/lower game, Bella!"

// Step 2: Prompt for lower and upper bounds with input validation

REPEAT

DISPLAY "Enter the lower bound: "

INPUT lower\_bound

DISPLAY "Enter the upper bound: "

INPUT upper\_bound

IF lower\_bound >= upper\_bound THEN

DISPLAY "The lower bound must be less than the upper bound."

END IF

UNTIL lower\_bound < upper\_bound

// Step 3: Generate a random number between lower\_bound and upper\_bound

SET random\_number TO RANDOM (lower\_bound, upper\_bound)

// Step 4: Initialize game state

SET game\_over TO FALSE

// Note: Assume all inputs are valid numbers for simplicity

// Step 5: Game loop to prompt for guesses with input validation

REPEAT

DISPLAY "Great, now guess a number between ", lower\_bound, " and ", upper\_bound, ": "

INPUT guess

// Validate guess is within bounds

IF guess < lower\_bound OR guess > upper\_bound THEN

DISPLAY "Nope, your guess must be between ", lower\_bound, " and ", upper\_bound, "."

ELSE

// Decision branching based on guess

IF guess < random\_number THEN

DISPLAY "Nope, too low."

ELSE IF guess > random\_number THEN

DISPLAY "Nope, too high."

ELSE

DISPLAY "You got it!"

SET game\_over TO TRUE

END IF

END IF

UNTIL game\_over IS TRUE

END

**Brief Explanation**

This is a pseudocode where the higher/lower Game where a player guesses a number between a user-defined lower and upper bound:

**Welcome:** Displays a welcome message for Bella once the game is started.

**Get Bounds** prompts the user to provide a lower and upper bound, looping until the lower bound is less than the upper bound (input validation).

**Random Number**: Generates a random number between the bounds.

**Game Loop**: Repeatedly asks the user for a guess, ensuring it’s within the bounds (input validation).

If the guess is too low, it outputs "Nope, too low."

If too high, it outputs "Nope, too high."

If correct, it outputs "You got it!" and ends the game.

**Loop Control**: Continues prompting for guesses until the correct number is guessed.

The pseudocode uses loops for input validation and gameplay, and decision branching to provide feedback on guesses.