

python3 student_no.py # takes inputs from the terminal and displays results (if any).

python3 student_no.py<input1.txt # it takes inputs from input1.txt and displays the result in the terminal.

python3 student_no.py<input1.txt>myoutput.txt # it takes the inputs from input1.txt and creates the myoutput.txt document and writes the result here.

diff output1.txt myoutput.txt # compares the document output1.txt with the document myoutput.txt.

diff --ignore-all-space output1.txt myoutput.txt # compares the document output1.txt with the document myoutput.txt without ignoring the spaces.

QUESTION 1

Implement a **RECURSIVE** function

computer_guess_recursive(low, high, target, guesses_left) that does the following:

Initial Conditions:

- The computer will be given a range of numbers (e.g., from 1 to 100).
- A target number between low and high is randomly selected.
- The computer has a total of 7 attempts to guess the number correctly.

Feedback Mechanism:

- If the computer guesses correctly, the program will display a message saying the computer has guessed the target correctly.
- If the computer's guess is too high, it will adjust the upper bound
- If the computer's guess is too low, it will adjust the lower bound
- After each guess, the updated range and the computer's guess will be printed.
- If no valid guesses remain (i.e., the guess count reaches zero), the program will inform the user that the computer failed to find the target number.

Error Handling:

- Handle invalid values gracefully using try/except blocks. For example, if a non-integer value is passed to the function, the program should print an error message.

def computer_guess_recursive(low, high, target, guesses_left)

Parameters:

low (int): The lower bound of the range.

high (int): The upper bound of the range.

target (int): The target number that the computer is trying to guess.

guesses_left (int): The number of remaining guesses.

Note: It is mandatory to use a **RECURSIVE FUNCTION**.

Output 1

Computer has chosen the target number to be 4.

Computer's guess: 65, current range: [1, 100], remaining guesses: 6

Computer's guess of 65 is too high, the target is smaller.

Computer's guess: 1, current range: [1, 64], remaining guesses: 5

Computer's guess of 1 is too low, the target is larger.

Computer's guess: 27, current range: [2, 64], remaining guesses: 4

Computer's guess of 27 is too high, the target is smaller.

Computer's guess: 25, current range: [2, 26], remaining guesses: 3

Computer's guess of 25 is too high, the target is smaller.

Computer's guess: 6, current range: [2, 24], remaining guesses: 2

Computer's guess of 6 is too high, the target is smaller.

Computer's guess: 5, current range: [2, 5], remaining guesses: 1

Computer's guess of 5 is too high, the target is smaller.

Computer's guess: 3, current range: [2, 4], remaining guesses: 0

Computer's guess of 3 is too low, the target is larger.

You are out of guesses! The computer couldn't guess the target.

Output 2

Computer has chosen the target number to be 11.

Computer's guess: 21, current range: [1, 100], remaining guesses: 6

Computer's guess of 21 is too high, the target is smaller.

Computer's guess: 4, current range: [1, 20], remaining guesses: 5

Computer's guess of 4 is too low, the target is larger.

Computer's guess: 15, current range: [5, 20], remaining guesses: 4

Computer's guess of 15 is too high, the target is smaller.

Computer's guess: 11, current range: [5, 14], remaining guesses: 3

**Computer correctly guessed at 4 guesses the target!
Your target number is: 11**