

# Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT114-002-S2024/it114-number-guesser-4/grade/ob75>

IT114-002-S2024 - [IT114] Number Guesser 4

## Submissions:

Submission Selection

1 Submission [active] 2/12/2024 5:47:47 PM

## Instructions

^ COLLAPSE ^

- 1 .Create the below branch name
- 2 .Implement the NumberGuess4 example from the lesson/slides
  - 1 <https://gist.github.com/MattToegel/aced06400c812f13ad030db9518b399f>
- 3 Add/commit the files as-is from the lesson material (this is the base template). You may want to push this commit so you can open the pull request and keep it open.
- 4 .Pick two (2) of the following options to implement
  - 1 .Display higher or lower as a hint after a wrong guess (only after a wrong guess that doesn't roll back the level)
  - 2 .Implement anti-data tampering of the save file data (reject user direct edits)
  - 3 Add a difficulty selector that adjusts the max strikes per level (i.e., "easy" 10 strikes, "medium" 5 strikes, "hard" 3 strikes)
  - 4 .Display a cold, warm, hot indicator based on how close to the correct value the guess is (example, 10 numbers away is cold, 5 numbers away is warm, 2 numbers away is hot; adjust these per your preference) Only display this when the wrong guess doesn't roll back the level
  - 5 Add a hint command that can be used once per level and only after 2 strikes have been used that reduces the range around the correct number (i.e., number is 5 and range is initially 1-15, new range could be 3-8 as a hint)
  - 6 .Implement separate save files based on a "What's your name?" prompt at the start of the game (each person gets their own save file based on user's name)
- 5 .Fill in the below deliverables
- 6 .Save changes and export PDF
- 7 .Git add/commit/push your changes to the HW branch
- 8 .Create a pull request to main
- 9 .Complete the pull request (don't forget to locally checkout main and pull changes to prep for future work)
- 10 Upload the same PDF to Canvas

Branch name: M3-NumberGuesser-4

Tasks: 7 Points: 10.00

## Implementation 1 (4 pts.)

^ COLLAPSE ^

## Task #1 - Points: 1

Text: Chosen Option and Details

## Checklist

\*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Mention which option you picked
<input type="checkbox"/> #2	1	Explain the logic of how you solved/implemented the chosen option (concrete details). Explain how the code works, don't just paste code snippets

Response:

I choose option 1 displaying higher or lower as a hint after a wrong guess and does only gives this hint if the guess does not rolls back the level. The way I solved this problem was by implementing an if and else if statement into the processGuess method which was in charge of deciding if you lose or win based on your guess. An else statement was put as opposite to if strikes was  $\geq$  maxStrikes so that if it doesn't roll back the level then make it do the else statement which was if guess < number then give a hint saying its higher and vice versa if guess > number then give then hint saying its lower.

## Task #2 - Points: 1

Text: 2+ Screenshots of code and demo

## Checklist

\*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show implementation working by running the program
<input type="checkbox"/> #2	1	Clearly caption the screenshot of what you're showing
<input type="checkbox"/> #3	1	The code screenshot(s) clearly show the code specific to the feature
<input type="checkbox"/> #4	1	A comment with the UCID/date is visible near the code change(s)

Task Screenshots:

☐ Large Gallery

Checklist Items (0)



Checklist Items (0)

The else statement code right below UCID and under the

The demo showing higher and lower hints as you guess

lose() condition in the block contains also the if and else if statement.

with a number and goes according to that number.



Checklist Items (0)

The demo showing it does not give the hint on the last guess you get wrong.

## Implementation 2 (4 pts.)

^ COLLAPSE ^

### Task #1 - Points: 1

Text: Chosen Option and Details

#### Checklist

\*The checkboxes are for your own tracking

#	Points	Details
<input checked="" type="checkbox"/> #1	1	Mention which option you picked
<input checked="" type="checkbox"/> #2	1	Explain the logic of how you solved/implemented the chosen option (concrete details). Explain how the code works, don't just paste code snippets

#### Response:

I picked option 3 for my second implementation in which the user selects a difficulty between Easy(10 strikes), Medium(5 strikes), and Hard(3 strikes). I first went to the start method to put a string code telling the user to pick a difficulty between the 3 I mentioned and telling them how many strikes each difficulty has. Once they type the difficulty, it would pass through to that difficulty with with limited strikes and would then start the game. The same code that was used if the user wanted to quit the game by typing 'quit'. This was done for each of the difficulties using and if statement for the first than else if for if they typed a different difficulty. maxStrikes was then called back and changed to be a certain amount for each of the difficulties for example, for hard, maxStrikes was changed to 3 in this case.

### Task #2 - Points: 1

Text: 2+ Screenshots of code and demo

#### Checklist

\*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Show implementation working by running the program
<input type="checkbox"/> #2	1	Clearly caption the screenshot of what you're showing

<input type="checkbox"/> #3	1	The code screenshot(s) clearly show the code specific to the feature
<input type="checkbox"/> #4	1	A comment with the UCID/date is visible near the code change(s)

#### Task Screenshots:

☐ Large Gallery



Checklist Items (0)



Checklist Items (0)

My code block under UCID and ends before the loadState(); contains an if statement with 2 else if statements.

This demo is showing the 'Easy' mode with 10 strikes.



Checklist Items (0)



Checklist Items (0)

This demo is showing the 'Medium' mode with 5 strikes.

This demo is showing the 'Hard' mode which is with 3 strikes.

Misc (2 pts.)

^ COLLAPSE ^

Task #1 - Points: 1

Text: Reflection

#### Checklist

\*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Example prompts: Learn anything new? Face any challenges? How did you overcome and issues?
<input type="checkbox"/> #2	1	At least a few logical sentences related to the assignment.

#### Response:

When I was doing the implementation of option 3, I ran into the problem of the code asking the user to select a difficulty and enter a number at the same time and the way I fixed it was that I moved the loadState() function to under the block of code I created so that it would ask what difficulty wants to play on first and then once chosen, it would start the game. There was a part where the game wasn't reading the input of difficulty but that was due to my error of not including a nextLine() function and then added that to be set to the difficulty variable.



COLLAPSE

Task #2 - Points: 1

Text: Pull Request URL

**Details:**

URL should end with /pull/# where the # is the actual pull request number.

URL #1

<https://github.com/OmarBarrera1/ob75-IT114-002/pull/3>

COLLAPSE

Task #3 - Points: 1

Text: Waka Time (or related) Screenshot

**Checklist**

\*The checkboxes are for your own tracking

#	Points	Details
<input type="checkbox"/> #1	1	Screenshot clearly shows what files/project were being worked on (the duration of time doesn't correlated with the grade for this item)

Task Screenshots:

Large Gallery



Checklist Items (0)

Wake time was not calculating time I worked since I worked on this a few days before taking screenshots. This shows the time I worked for on the day I did this which is the 7th.

End of Assignment