

# Submission Worksheet

CLICK TO GRADE

<https://learn.ethereallab.app/assignment/IT114-003-F2024/it114-module-3-number-guesser-4/grade/js2637>

Course: IT114-003-F2024

Assignment: [IT114] Module 3 Number Guesser 4

Student: Jack S. (js2637)

## Submissions:

Submission Selection

1 Submission [submitted] 9/27/2024 5:13:43 PM

## Instructions

^ COLLAPSE ^

Overview Video: <https://youtu.be/ej6lWrg9XjE>

1. Create the below branch name
2. Implement the NumberGuess4 example from the lesson/slides
  1. <https://gist.github.com/MattToegel/aced06400c812f13ad030db9518b399f>
  2. Add/commit the files as-is from the lesson material (this is the base template).
  3. Push the changes to the HW branch and create a pull request to keep open until this assignment is done
3. 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)
4. Fill in the below deliverables
5. Save changes and export PDF

6. Git add/commit/push your changes to the HW branch
7. Create a pull request to main (if not done so before)
8. Complete the pull request (don't forget to locally checkout main and pull changes to prep for future work)
9. Upload the same PDF to Canvas

Branch name: M3-NumberGuesser-4

Group



Group: Implementation 1

Tasks: 1

Points: 4

^ COLLAPSE ^

Task



Group: Implementation 1

Task #1: Implementation Evidence

Weight: ~100%

Points: ~4.00

^ COLLAPSE ^

**i** Details:

Code screenshots must have ucid/date shown as a comment in the code.

Explanations must be your own words describing the logic and how the solution code solves the problem. ⌵

Columns: 1

Sub-Task



Group: Implementation 1

Task #1: Implementation Evidence

Sub Task #1: Mention which option you picked and how you solved it

## ≡ Task Response Prompt

*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:

Display higher or lower as a hint after a wrong guess (only after a wrong guess that doesn't roll back the level). I just added a simple if statement in the processGuess method so that if the guess is lower than the number it will tell the user to guess higher and vice versa.

Sub-Task



Group: Implementation 1

Task #1: Implementation Evidence

100%

Sub Task #2: Add screenshots of the coded solution (ucid/date must be visible)

## Task Screenshots

Gallery Style: 2 Columns

4

2

1

```
//js2637 9/27/2024
//worked with my brother Eric from it114-005
else if (guess>number)
{
    System.out.println("guess Lower ");
    System.out.println (warm(guess, number));
}
<- #141-145 else if (guess>number)
else if (guess<number)
{
    System.out.println("guess Higher ");
    System.out.println (warm(guess, number));
}
```

Code

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Sub-Task

Group: Implementation 1

Task #1: Implementation Evidence

Sub Task #3: Show implementation working by running the program

100%

## Task Screenshots

Gallery Style: 2 Columns

4

2

1

```
Welcome to level 1.
I picked a random number between 1-10, let's see if you can guess.
Type a number and press enter
3
You guessed 3
That's right!
Welcome to level 2.
I picked a random number between 1-15, let's see if you can guess.
Type a number and press enter
4
You guessed 4
That's wrong
guess higher
You are cold (a least 10 units away from the number)
Type a number and press enter
7
You guessed 7
That's wrong
guess higher
You are warm (5 units away from the number)
Type a number and press enter
9
You guessed 9
That's wrong
guess higher
You are hot (2 units away from the number)
Type a number and press enter
10
```

code running

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

End of Task 1

End of Group: Implementation 1

Task Status: 1/1

Group

Group: Implementation 2

Tasks: 1

100%

Points: 4

^ COLLAPSE ^

## Task

100%

Group: Implementation 2

Task #1: Implementation Evidence

Weight: ~100%

Points: ~4.00

^ COLLAPSE ^

## Details:

Code screenshots must have ucid/date shown as a comment in the code.

Explanations must be your own words describing the logic and how the solution code solves the problem.



Columns: 1

## Sub-Task

100%

Group: Implementation 2

Task #1: Implementation Evidence

Sub Task #1: Mention which option you picked and how you solved it

## Task Response Prompt

*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:

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. I created the method called warm(), which takes two int values then I used the math absolute operator to find the distance between the guess and the answer. Then, the if statement is created to return a String based on the distance.

## Sub-Task

100%

Group: Implementation 2

Task #1: Implementation Evidence

Sub Task #2: Add screenshots of the coded solution (ucid/date must be visible)

## Task Screenshots

Gallery Style: 2 Columns

4

2

1





Warm Code

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

Sub-Task

Group: Implementation 2

Task #1: Implementation Evidence

Sub Task #3: Show implementation working by running the program

100%

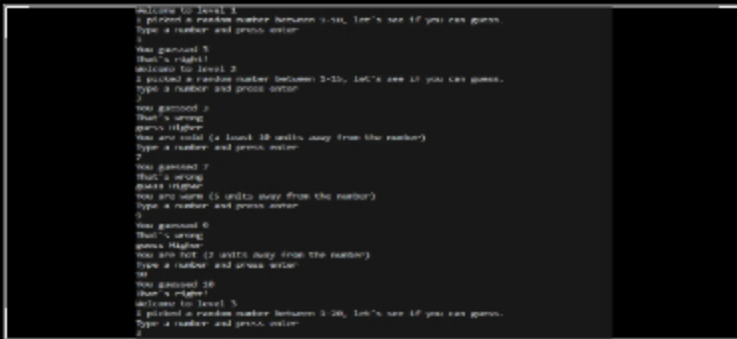
## Task Screenshots

Gallery Style: 2 Columns

4

2

1



code running

Caption(s) (required) ✓

Caption Hint: *Describe/highlight what's being shown*

End of Task 1

End of Group: Implementation 2

Task Status: 1/1

Group

Group: Misc

Tasks: 3

Points: 2

100%

⬆️ COLLAPSE ⬆️

Task

Group: Misc

Task #1: Reflection

Weight: ~33%

Points: ~0.67

100%

⬆️ COLLAPSE ⬆️

**Sub-Task**

Group: Misc

Task #1: Reflection

Sub Task #1: Learn anything new? Face any challenges? How did you overcome any issues?

100%

## Task Response Prompt

*Provide at least a few logical sentences*

Response:

I learned to save my file before Javac because I was having issues with my code giving errors even when I fixed it, then realized that it was because my file wasn't saved. I learned that the math absolute operator is useful for finding distance because at first, I did an if statement like `(guess-number<=2 || number-guess<=2)`, this code didn't work out because if you think about it isn't the code always going to be true, so I realized that I should find the absolute value to find the distance.

End of Task 1

**Task**

Group: Misc

Task #2: Pull Request URL

Weight: ~33%

Points: ~0.67

100%

^ COLLAPSE ^

**Details:**URL should end with `/pull/#` where the `#` is the actual pull request number.

## Task URLs

URL #1

<https://github.com/Jackshii/Js2637-IT114-003/pull/6>

URL

<https://github.com/Jackshii/Js2637-IT114-003/p>

End of Task 2

**Task**

Group: Misc

Task #3: Waka Time (or related) Screenshot

Weight: ~33%

Points: ~0.67

100%

^ COLLAPSE ^

**Checklist**

\*The checkboxes are for your own tracking

#

Details

#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

Gallery Style: 2 Columns

4

2

1



WakaTime

End of Task 3

End of Group: Misc  
Task Status: 3/3

End of Assignment