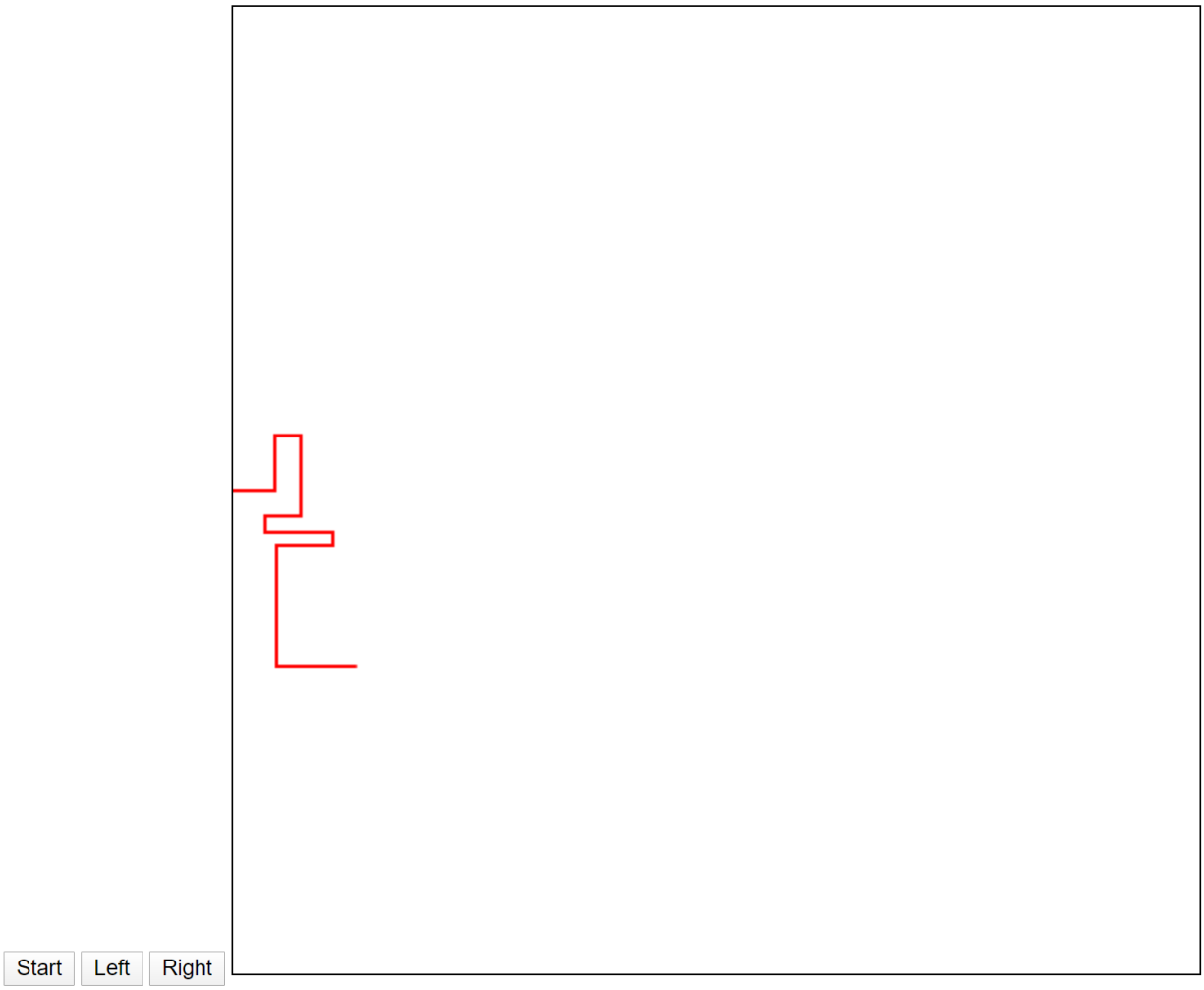
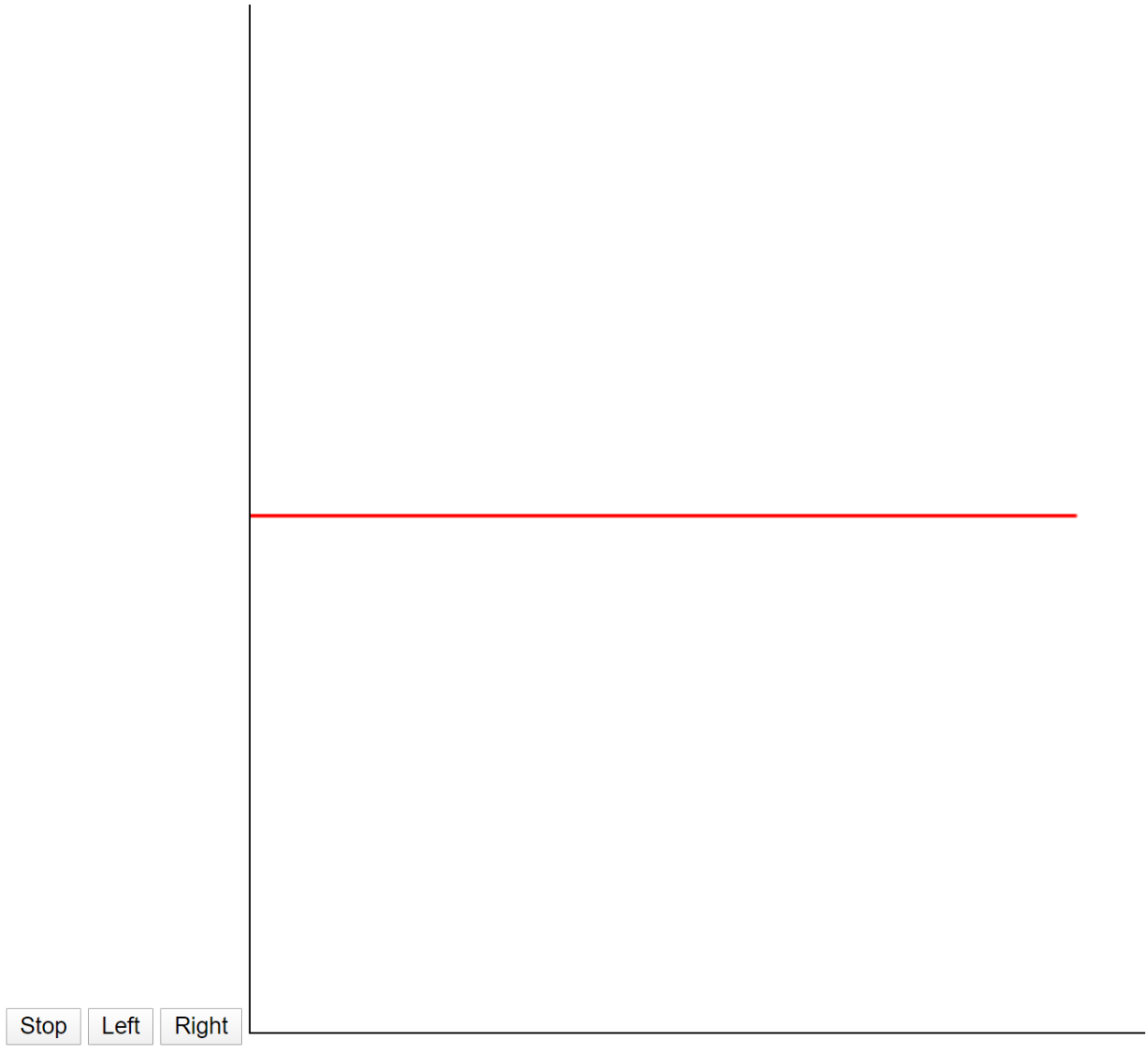
Hw 4 Report

Vismay Gehlot

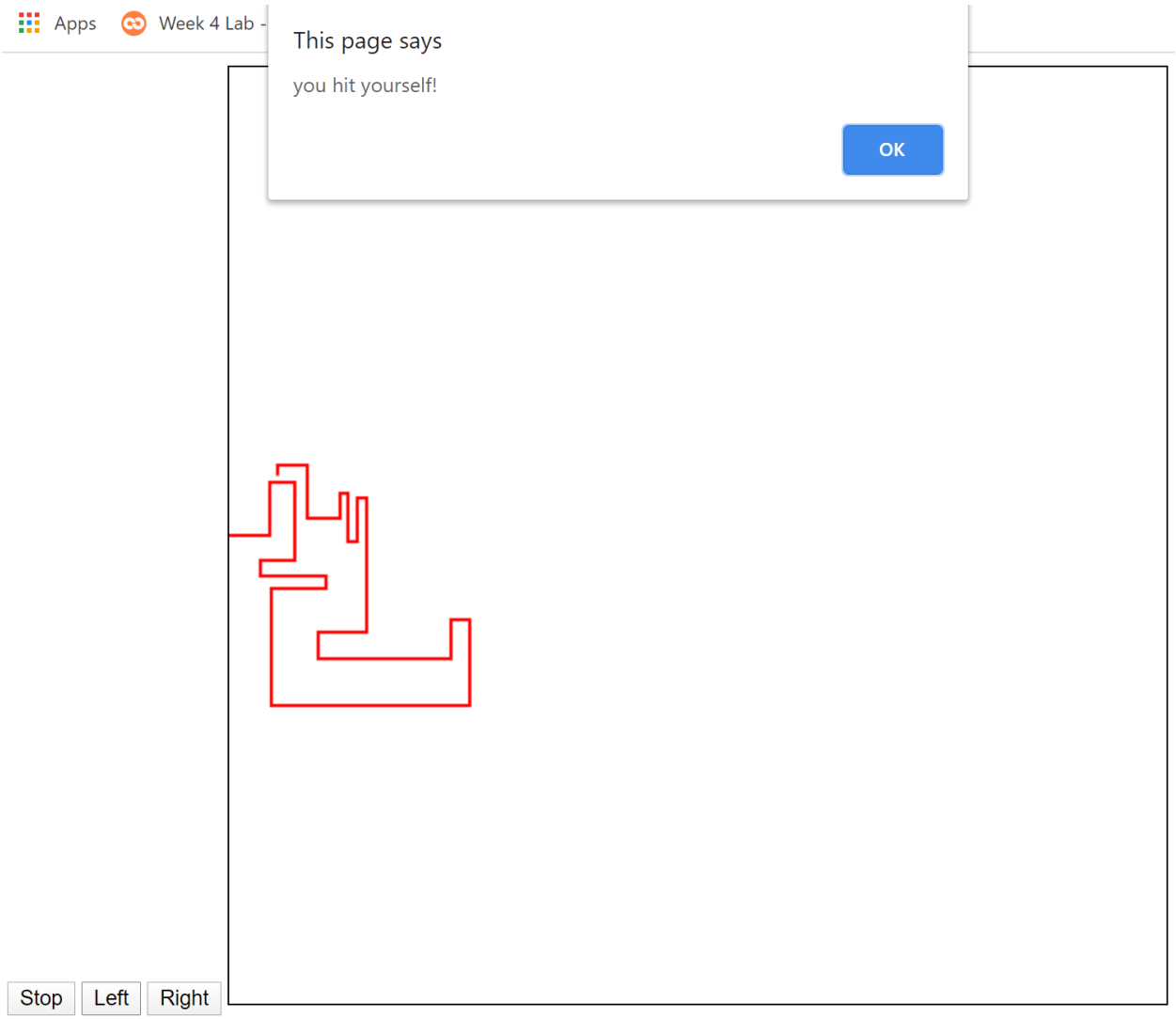
For the snake project, I used a timer to create the snake itself. The timer schedule itself decided how often the snake would move forward. For the turn functions, I used a series of if statements that would first check what direction the snake was currently headed in and then if a right or left button was clicked, the snake would turn to the desired direction depending on where it was currently facing. In order to see when the snake has died, I used the borders of the game itself as bounds to dictate whether or not the snake has hit a wall. If the x or y coordinates are ever out of the set bounds, then the game stops and an alert message is displayed. To check that the snake hadn't gone over itself by accident, I had to check every iteration to make sure it hasn't collided with another part of itself. This was achieved by seeing if the next block in the way of the snake is the same color as the body. If it is, then another death message is sent through alert and the user must refresh the page to play again.



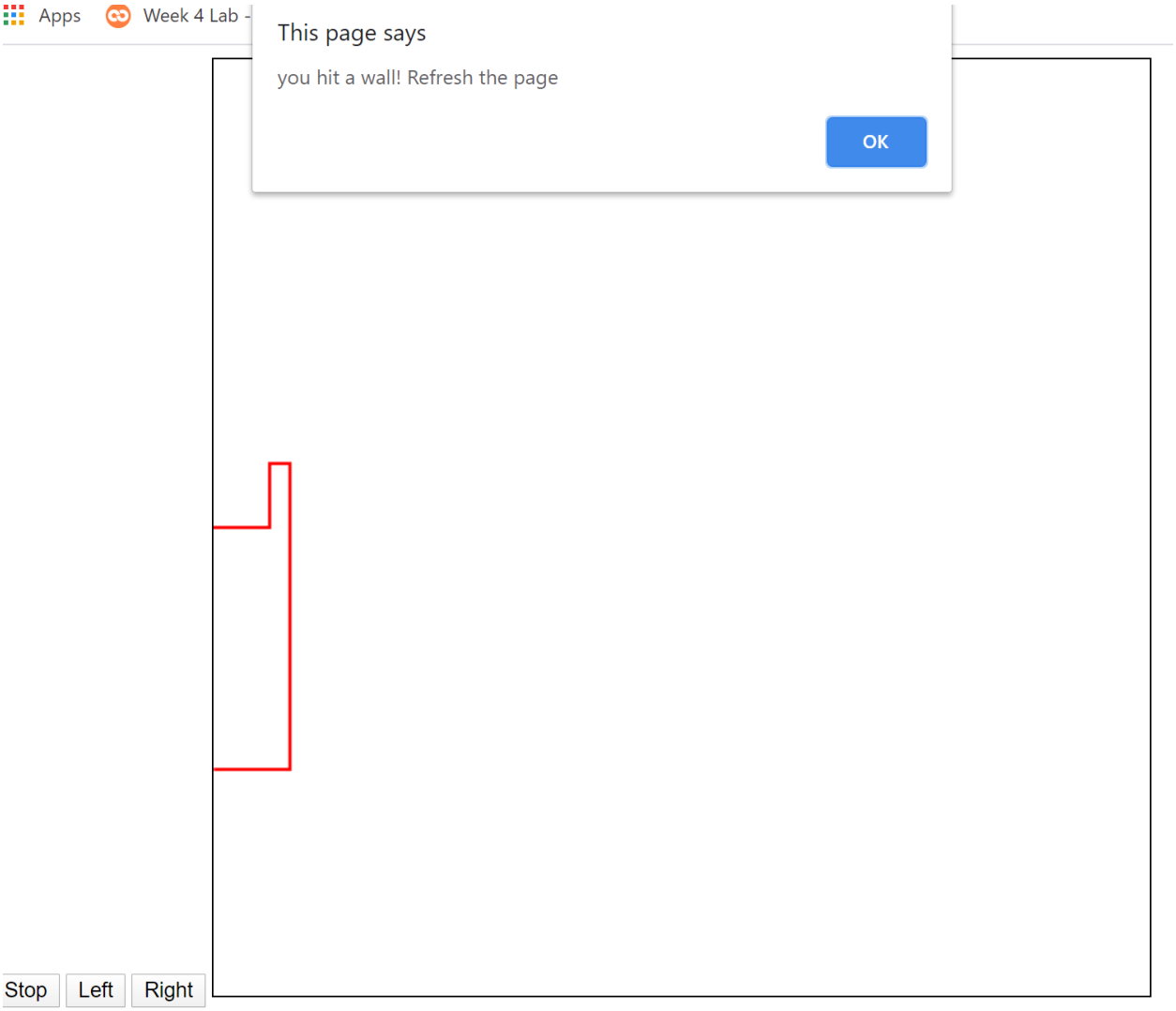
Pause function while paused, the button says start to prompt the user to resume the game.



As seen above, while the game is still running, the start button changes to stop, so that the user knows that clicking it will pause the game/



If the snake hits itself, game ends and alert message pops on screen.



If the user hits a wall, the game ends and the user is alerted.

TASK 2:

```
C:\Users\visma\OneDrive\Desktop\Iowa State Stuffs\Spring 2020\319\Homework 4>node hw4.js
1st Number: 1
2nd Number: 2
3rd Number: 3
4th Number: 12321
Factorial of the 1st number is = 1
The sum of all digits of the 2nd number is = 2
The reverse of the 3rd number is = 3
Is the 4th number a palindrome (True/False)? = true

C:\Users\visma\OneDrive\Desktop\Iowa State Stuffs\Spring 2020\319\Homework 4>node hw4.js
1st Number: 5
2nd Number: 1234
3rd Number: 1234567
4th Number: 1223443321
Factorial of the 1st number is = 120
The sum of all digits of the 2nd number is = 10
The reverse of the 3rd number is = 7654321
Is the 4th number a palindrome (True/False)? = false

C:\Users\visma\OneDrive\Desktop\Iowa State Stuffs\Spring 2020\319\Homework 4>
```

For task 2, the inputs and outputs are shown above. In the above screenshot, the program has been run twice.

For the first task, I used a recursive method to keep calling the factorial method. This would keep recursing until the final number was a 0, then return the summation of all values.

For the second task, I changed the type of the number to a string, then separated each value in the string as its own. After this, I changed each value back to a number and stored it in an array and added them together to return the final output.

For the third task, I once again converted the number type to a string, then used the split method and then the reverse method to flip the number. After this, I typecasted it back as an integer and returned the value.

For the fourth task, I simply called the flipping method that I used for task 3, ran it on the number input, and then compared the flipped number with its original. If they were equal, then this number was a palindrome and the function returned true. If not, then it returned false.