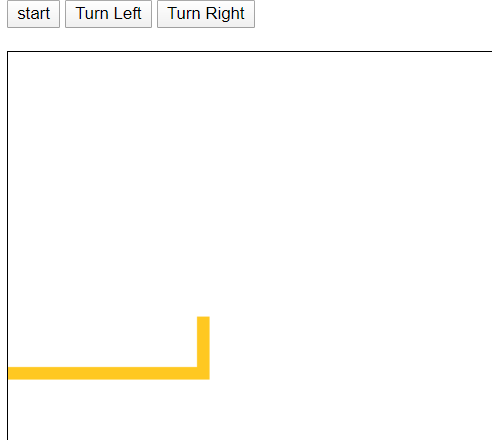
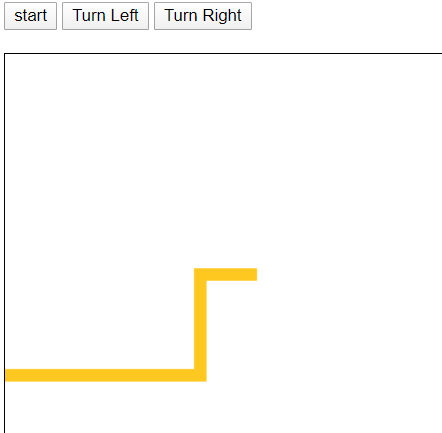
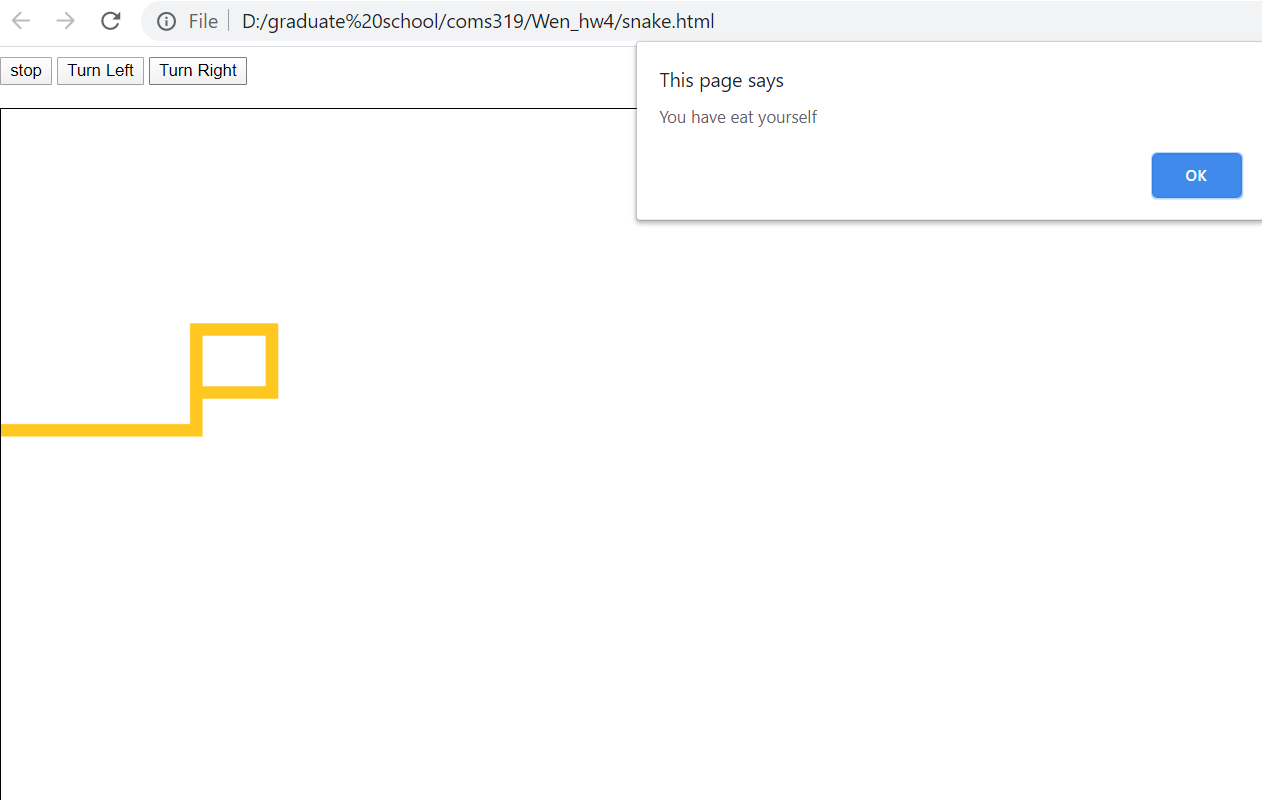
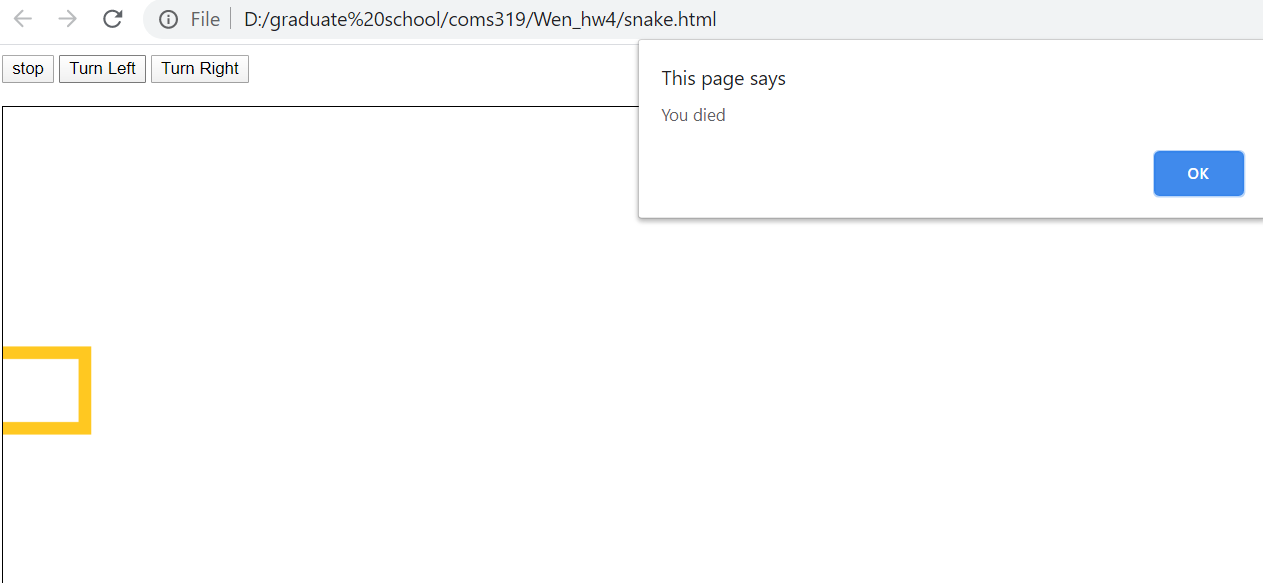
**CS319 HW4 2/18/2020 Zhanghao Wen**

**Announcement**: This is hw#4 and for task 2 I named file hw4.js instead of hw3.js (named from instruction) as well as HW4.zip instead of HW3.zip.

My code satisfies all requirements without any additional improvement.

# Task 1: Event Handling

## Screenshot

## Solution Approach

Based on the screenshots above, my code is able to:

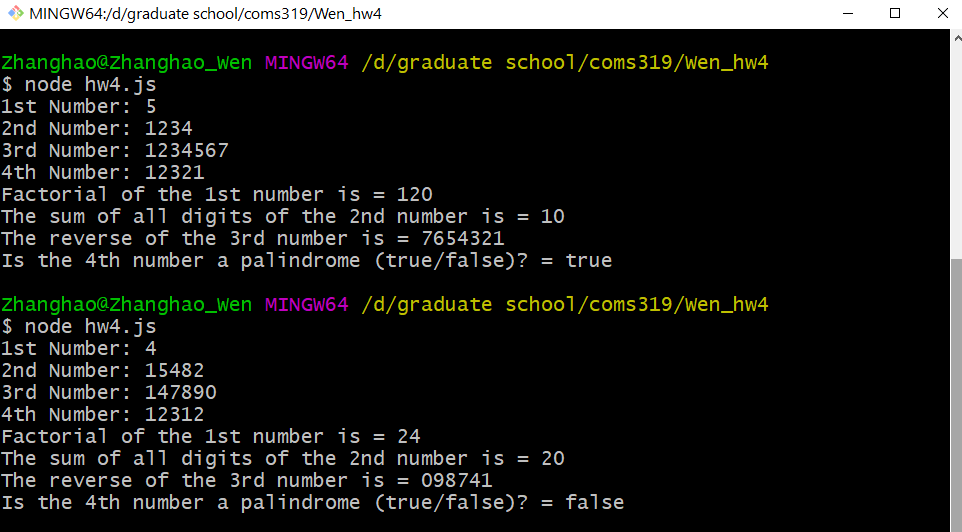
* The line can go over any upcoming path
* The line can bend left when left button clicked
* The line can bend right when right button clicked
* The line can stop when:
  + Stop button pressed
  + Snake head touch its body
  + Snake head touch edge of rectangle

**Procedure**:

* Create buttons and canvas in html.
* When start button clicked, call start() function which will set timer and call run() function every 0.5 seconds.
* Set currentDirection globally and set it to “RIGHT” at first, and then it will change based on button clicked.
* In run() function, check whether snake head touch edge or itself, if yes then terminate and send alert.
* Run() function calls draw() function which will fill the little rectangle which forms the path eventually and draw(direction) take currentDirection as parameters so it can draw little rectangle according to the direction.

# Task 2: NodeJS

## Screenshot:



## Solution Approach

Based on the screenshots above, my code is able to:

* Take four integer numbers as input
* Calculate the factorial of the first number
* Calculate the sum of all the digits of the second number
* For the third number given as an input, show the reversed number as an output
* For the fourth number given as an input, check whether that number is a Palindrome or not

**Procedure**:

* Create js file and npm install readline-sync
* Use 4 variables to store 4 inputs from user
* Create factorial function, sum digit function, reverse function and palindrome function
* Process inputs to those function and printout.
* The function details are simple, and they are in the code.