CPSC 1045 – Lab #6 (TEAM LAB OPTION)

Objective(s):

1. Learn JavaScript events' concepts and terms
2. Write event handlers
3. Learn Event Propagation
4. Learn JavaScript arrays and array methods
5. Evaluate JavaScript code fragments that include arrays
6. Write JavaScript code fragment using arrays
7. Create a webapp using HTML/JavaScript
    a. Implement events
    b. Implement arrays

Resource(s):

- http://www.w3schools.com/html/default.asp
- http://www.w3schools.com/js/default.asp
- http://www.w3schools.com/canvas/default.asp
- CPSC-1045-Lab-6-Question-III-KEY.zip
- Web Browser and a computing device (campus or personal)
- Lecture and Lab notes
- Prior Labs and Worksheets

Grade Value:

- Tasks I, II, III, IV, V, and VI - **50 Points**
- Task VII – **150 Points**

Assigned Task(s):

I. JavaScript Events Concepts and Terms **(9 Points)**
    a. What is a JavaScript event?
    b. What is a HTML node?
    c. What is a handler used for?
    d. Name the four mouse movement events covered thus far.
    e. What is the load event used for?
    f. What event do you write a handler for to listen for mouse clicks?
    g. What is event.which used for?
    h. What is an event object?
    i. Name the three key events covered thus far.
    j. What does event propagation mean?
    k. Describe why the scroll event may be useful in a site/app/page?
    l. What is a bug?
    m. Why use try/catch for debugging?

    n. What is the purpose of using assertions for debugging?

o. What is the keycode for the letter “A”?
p. What is a modifier key?
q. When might you want to handle focus events?
r. Give an example of when a handler must be implemented for the load event.

II. Write event handlers **(6 Points)**

a. Write an event handler for a “mousemove” event on a canvas (myCanvas). When this event occurs, output to the console the message “mouse moved”. **(2 Points)**
b. Write an event handler for a “keydown” event on the webpage. When the Ctrl-L key is pressed, output to the console the message “Ctrl-L pressed”. **(2 Points)**
c. Write an event handler when the “button2” on a canvas is pressed. When this button is pressed, call a function named button2Pressed that will output to the console the message “button 2 disabled”, and then remove this event handler. **(2 Points)**

III. Learn Event Propagation events **(10 Points)**

Download **CPSC-1045-Lab-6-Question-III-KEY.zip** from D2L. Launch this webapp, and then answer the following:

a. What happens when you click “Default Text” on the page? **(1 Point)**
b. Refresh the webapp – then what happens when you click the button? **(1 Point)**
c. Review a. and b. again – how/why are they different? **(2 Points)**
d. Refresh the webapp – then what happens when you click the canvas (the smaller box)? **(1 Point)**
e. Refresh the webapp – then what happens when you click the Heading (**CPSC 1045 – Lab #6 – Question – III**)? **(1 Point)**
f. Review d. and e. again – how/why are they different? **(2 Points)**
g. Refresh the webapp – then what happens when you click just to the left or right of the large box – somewhere midway between the top and bottom of this box? **(1 Point)**
h. Refresh the webapp – then click outside the large box so that only the **html handler** text is displayed in the canvas? Hint – you will need click outside the <body> tag’s “region”. **(1 Point)**

Review the html file to understand Propagation of events, and how it has been coded to demonstrate event propagation.

IV. JavaScript Array Concepts and Terms **(6 Points)**

a. What is an array?
b. How do we access an element in the array?
c. Write the JS statement to change the fifth element on a one-dimensional array called numberList to the value of -5.
d. Which method do we use to add an element to a one-dimensional array?
e. Which method do we use to remove the last element from a one-dimensional array?
f. What array method is used to take away two elements from a one-dimensional array?
g. What array method is used to insert multiple elements anywhere in a one-dimensional array?
h. What does the concat() method do for single dimension arrays?
i. How do we create an empty array?

j. What data can an array store?

k. Can an array hold multiple types at once?

l. What program structure have you learned thus far would you use to initialize an array to all zeros?

V. Evaluate JavaScript code fragments **(9 Points)**

a. Given the following array: **(1 Point)**

```
var arr = [ 'dog',  'cat', 'deer'];

var result = arr[0]+arr[2];
```

What is the value of **result**?

b. Given the following program, add the code necessary to assign **children** the concatenation of the arrays **girls** and **boys**. **(2 Points)**

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
var girls = ["Cecilie", "Lone"];
var boys = ["Emil", "Tobias", "Linus"];
var children = // add code here
document.getElementById("demo").innerHTML = children;
</script>

</body>
</html>
```

c. Analyze the following code fragment, and answer the questions below. **(2 Points)**

```
var arr = [ 20, 30 ];
for (var i = arr.length; i < 5; i += 1){
   arr[i] = Math.pow(i,2);
}
```

Will the above for loop execute without error?

If it does execute, what will be contained in the array **arr** after the loop terminates?

d. Analyze the following code fragment, and answer the questions below. **(2 Points)**

```
var arr = [ 10, 20, 30, 40, 50, 60, 70, 90 ];
var sum = 0;

for (var i = 1; i < 7; i += 1){
   sum = sum + arr[i];
}
```

What is the value **sum** when the loop terminates?

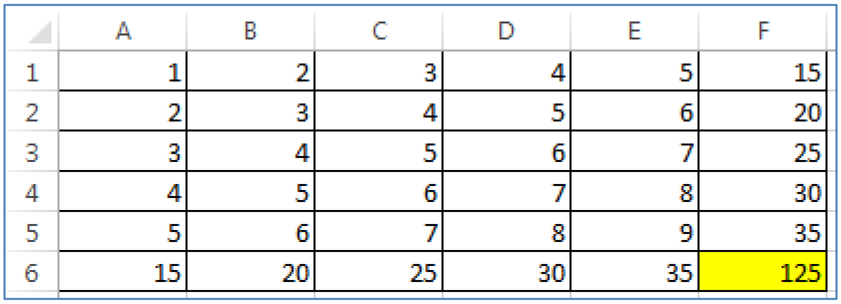
The sum after the loop terminates should be 370. Fix the necessary statements to ensure sum is 370.

e. Given the following code fragment, what is the output to the console after the completion of this code fragment? **(2 Points)**

```
// code fragment from inside main - start trace here
var  DL = 5;
var  d = [25.0, 9.0, 10.0, 25.0, 15.0];
var  mi = 0;
var  m = d[mi];
for (var  i = 1; i < DL; i++) {
   if  (d[i] < m) {
      mi = i;
      m = d[mi];
      console.log(m);
   }
}
console.log(“mi = “,mi,“ m = “,m);
// code fragment from main .. end trace here
```

VI. Write JavaScript code fragment using arrays **(10 Points)**

Given the following spreadsheet, write a function (called sumArray) that will receive an array, and a number representing the size of the array, and return the value of the sum indicated in yellow below.



	A	B	C	D	E	F
1	1	2	3	4	5	15
2	2	3	4	5	6	20
3	3	4	5	6	7	25
4	4	5	6	7	8	30
5	5	6	7	8	9	35
6	15	20	25	30	35	125

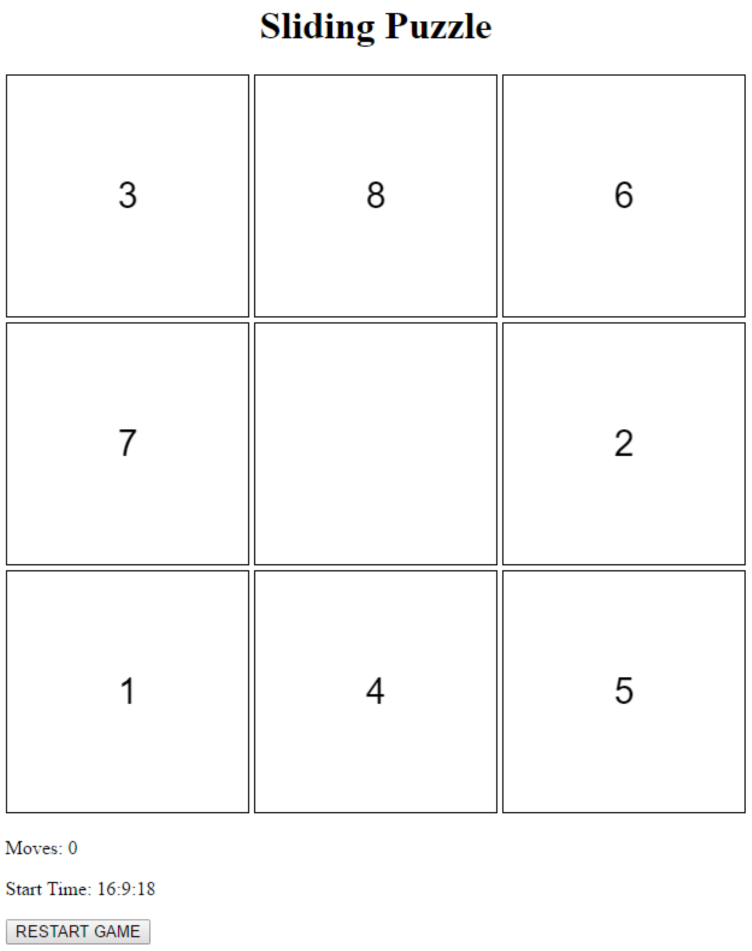
VII. Create a webapp using HTML/JavaScript **(150 Points)**

Write a webapp to implement a sliding puzzle. Here is the definition from Wikipedia: “A sliding puzzle, sliding block puzzle, or sliding tile puzzle is a tour puzzle that challenges a player to slide (frequently flat) pieces along certain routes (usually on a board) to establish a certain end-configuration. The pieces to be moved may consist of simple shapes, or they may be imprinted with colors, patterns, sections of a larger picture (like a jigsaw puzzle), numbers, or letters.”

Complete the following requirements for this webapp:

a. Create the board and use numbers as the pieces to be moved. **(5 Points)**
b. Use array(s) to store the values of the pieces and/or to store valid moves of pieces. You must implement at least one array. **(30 Points)**
c. Add a “moves” <p> tag to update moves made. **(5 Points)**
d. Add a “Start Time” <p> tag that will show the current hours:minutes:seconds when the webapp is first launched. **(5 Points)**
e. Add a “Restart Game” <button> tag that will restart the game when pressed (no warning necessary). **(5 Points)**

Your webapp should start up similar to the following screenshot:



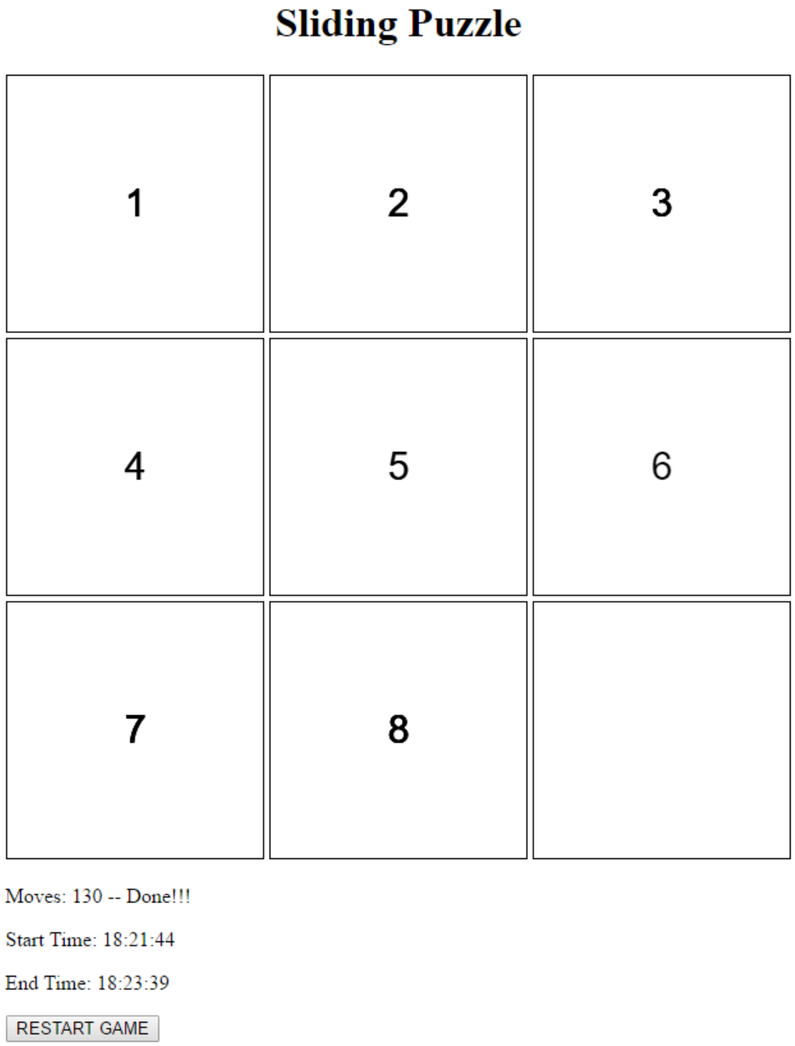
Sliding Puzzle

3	8	6
7		2
1	4	5

Moves: 0

Start Time: 16:9:18

f. Add an “End Time” <p> tag that will show the end time when the puzzle is solved. This tag is right below the “Start Time” <p> tag; but does not appear until the puzzle is solved. See example below. **(5 Points)**



Sliding Puzzle

1	2	3
4	5	6
7	8	

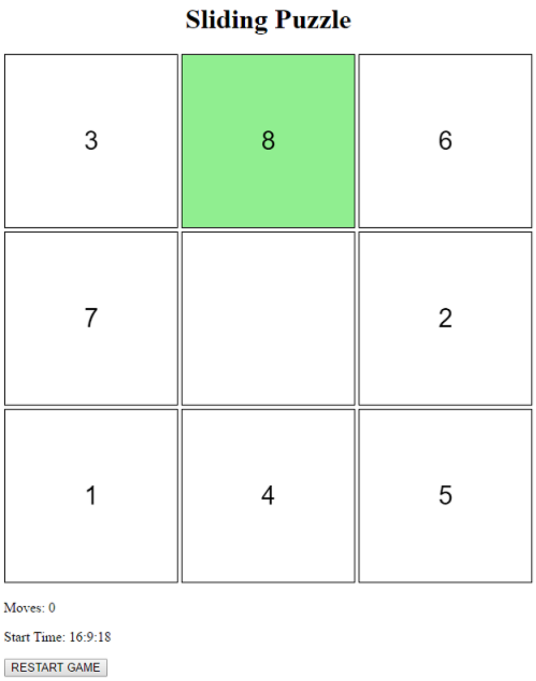
Moves: 130 -- Done!!!

Start Time: 18:21:44

End Time: 18:23:39

g. Write event handlers for the following:

- “Restart Button” **(15 Points)**
- Change the color of a piece to red if the mouse is over a piece that cannot be moved – reset the color back when the mouse moves away from that piece – see right screenshot below **(25 Points)**
- Change the color of a piece to green if the mouse is over a piece that can be moved – reset the color back when the mouse moves away from that piece – see left screenshot below **(25 Points)**

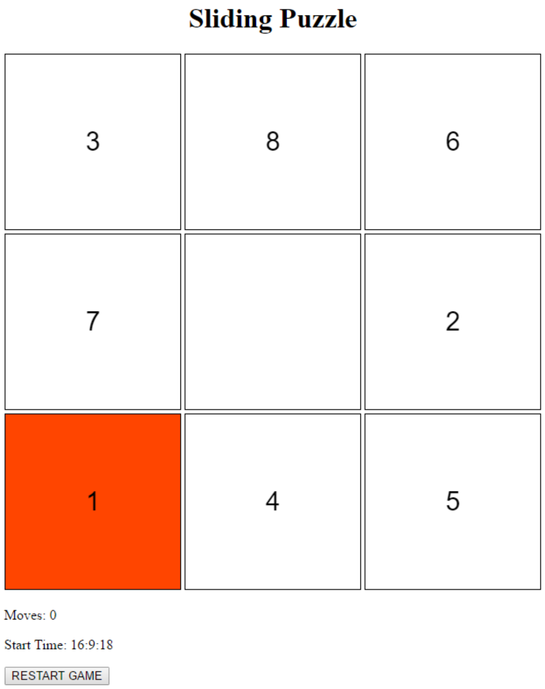


Sliding Puzzle

3	8	6
7		2
1	4	5

Moves: 0

Start Time: 16:9:18



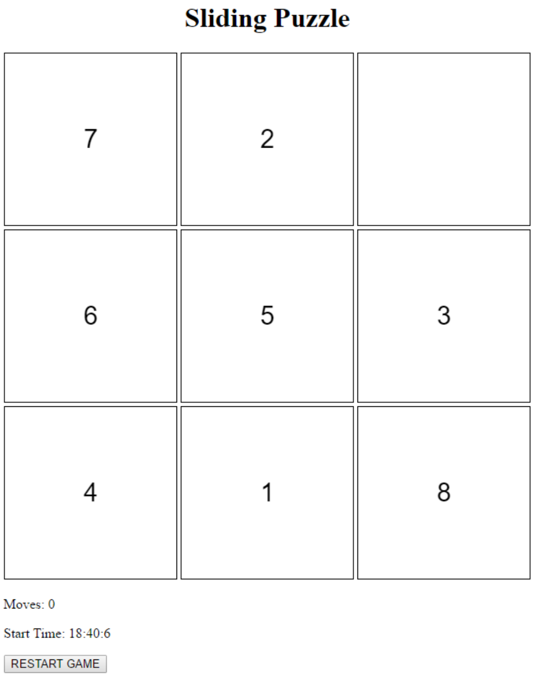
Sliding Puzzle

3	8	6
7		2
1	4	5

Moves: 0

Start Time: 16:9:18

- Process valid moves (clicks) to a piece – left screenshot below is start of puzzle; right screenshot below is the result of the first piece moved (2 from top-row-center to top-right) **(30 Points)**

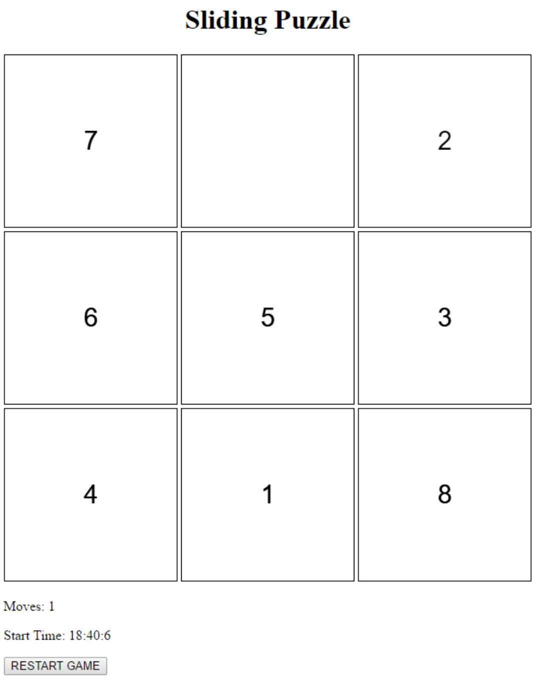


Sliding Puzzle

7	2	
6	5	3
4	1	8

Moves: 0

Start Time: 18:40:6



Sliding Puzzle

7		2
6	5	3
4	1	8

Moves: 1

Start Time: 18:40:6

Note – your board of pieces may be individual canvases or table cells.

Deliverable(s):

If you work with ONE other student as a team:

Submit one zipped file called **FirstName-LastName-SID-FirstName2-Lastname2-SID2-LAB6.zip** to D2L under the folder called Lab #6. This zip file will contain the following files:

FirstName-LastName-SID-FirstName2-Lastname2-SID2-LAB6.docx
This document will contain answers to Tasks I, II, III, IV, V, and VI above.

FirstName-LastName-SID-FirstName2-Lastname2-SID2-LAB6.HTML
This document will contain the webapp for Task VII above.

If you work by yourself:

Submit one zipped file called **FirstName-LastName-SID-LAB6.zip** to D2L under the folder called Lab #6. This zip file will contain the following files:

FirstName-LastName-SID-LAB6.docx
This document will contain answers to Tasks I, II, III, IV, V, and VI above.

FirstName-LastName-SID-LAB6.HTML
This document will contain the webapp for Task VII above.

Due Date:

See D2L. No late submissions accepted/graded.