



University of Maryland College Park

Dept of Computer Science

CMSC122, Final Exam

Spring 2011

First Name (PRINT): _____

Last Name (PRINT): _____

University ID: _____

I pledge on my honor that I have not given or received any unauthorized assistance on this examination.

Your signature: _____

Section (Circle One): **Nelson** (0101) → Mon/Wed

Fawzi (0201) → Tue/Thu

Instructions (Read the instructions before proceeding)

- This exam is a closed-book and closed-notes exam.
- Total point value is 100 points.
- Duration is 120 minutes.
- Please use a pencil to complete the exam.
- For those questions requiring JavaScript code just provide what should appear in between the `<script>` and `</script>` tags
- **WRITE NEATLY.** If we cannot understand your answer, we will not grade it (i.e., 0 credit).

#1	Miscellaneous		(20)
#2	HTML (Lists/Tables)		(10)
#3	CSS		(10)
#4	Loops		(10)
#5	One-Dimensional Arrays		(10)
#6	Two-Dimensional Arrays		(10)
#7	Randomize		(8)
#8	Name Generation		(12)
#9	Data Validation		(10)
Total			(100)

Problem 1, Miscellaneous (20 points)

1. (1 pt) An IP address represents a group of two or more machines on the internet. **T or F**
2. (1 pt) Data is transferred on the internet using packet switching. **T or F**
3. (1 pt) Global variables are variables defined inside of a function. **T or F**
4. (1 pt) The HTML goal is to describe the structure of a document. Presentation should be left to CSS. **T or F**
5. (1 pt) The submit button allows us to send form data to a server application. **T or F**
6. (1 pt) In HTML we can use <div> tags to identify the major sections (e.g., header, introduction, etc.) of the document. **T or F**
7. (1 pt) Infinite loops are only possible with while loops. **T or F**
8. (1 pt) Usability refers to how easy is the interaction between the user and a web site. **T or F**
9. (1 pt) A function can represent an event handler. **T or F**
10. (1 pt) The === (three =) is **more** strict than == (two =). **T or F**
11. (1 pt) One advantage of using functions is that it allows us to reuse code. **T or F**
12. (1 pt) We can validate the data associated with a form by recognizing the submit event. **T or F**
13. (1 pt) Which of the following is considered an event in JavaScript?
 - a) Generating a random number
 - b) Loading a web page
 - c) An infinite loop
14. (1 pt) The **get** method associated with forms reveals a lot about the information sent to the server when compared against the **post** method. **T or F**
15. (1 pt) DOM represents elements of a web page as a tree structure consisting of nodes. **T or F**
16. (1 pt) DHTML is the combination of HTML, CSS, and JavaScript. **T or F**
17. (1 pt) Name one html tag that is considered a block element.
18. (1 pt) Name one html tag that is considered an inline element.
19. (1 pt) What is a relational database?
20. (1 pt) Provide one example that illustrates the impact of software failures in society.

Problem 2, HMTL (Lists/Tables) (10 points)

1. Write the HTML code (only what goes in the <body></body> tags) that will define the following list. Shadows is a link to (<http://www.info.notreal>). You cannot type any numbers in your HTML.

- Concerts
 - Mozart
 - DC Orchestra
- Shows
 1. The Ghost
 2. [Shadows](#)

PUT HTML HERE

2. Write the HTML code (only what goes in the <body></body> tags) that will define the following table. The border value to use is 3. Make sure the header is in bold without using any bold tags (e.g., ,).

Road Name	Number
Beltway	495

PUT HTML HERE

Problem 3, CSS (10 points)

1. (2 pts) Define a CSS rule that associates the color blue with links that have not been visited.

2. (2 pts) Define a CSS rule that associates the color red with paragraphs in a document.

3. (2 pts) Define a CSS rule that defines 8em as the font size for div elements.

4. (2 pts) Define a CSS rule that defines green as the color of links that have been visited.

5. (1 pt) Comments in CSS are written using `/* */`
 - a) True b) False

6. (1 pt) A class selector allows us to apply the same CSS rules to different elements.
 - a) True b) False

Problem 4, JavaScript (Loops) (10 points)

Write a JavaScript program that prints a table with the square of **even** numbers between two specified values. The program will read the two values from the user by using prompt and the message “Enter Value”. For example, if the user enters 4 and 9 the table will be:

Number	Square
4	16
6	36
8	64

The program should work for values other than 4 and 9. Use the function Math.pow to compute squares (e.g., square of 3 → Math.pow(3,2))

Problem 5, JavaScript (One-Dimensional Arrays) (10 points)

Write a JavaScript function which has the following prototype: **function computeCeiling(data)**. The function will return a new array with the ceiling of the elements in the **data** array. The original array (data) should not be modified. For example, the following code fragment will show 3,7,10 in the alert box.

```
var a = [2.5, 6.8, 9.3];  
alert(computeCeiling(a));
```

The following information applies to this problem:

- You may not modify the array parameter.
- You only need to write the function (no need for main, <script>, <body>, etc.).
- The function does not read any data (i.e., it may not have any prompt statements).
- The function does not print any data (i.e., it may not have document.writeln or alert).
- The function must work for arrays of any length.
- Use the Math.ceil function to compute ceiling values (e.g., Math.ceil(2.5) → 3)

Problem 6, JavaScript (Two-Dimensional Arrays) (10 points)

Write a JavaScript function which has the following prototype: **function appendData(data)**. The function will return a String with the values in the **two-dimensional** array **data**. For example, the following code fragment will display “JoseMaryTomJackKyle” in the alert box:

```
var values=[["Jose", "Mary"],
            ["Tom", "Jack", "Kyle"]];

alert(appendData(values));
```

The following information applies to this problem:

- You may not modify the array parameter.
- You only need to write the function (no need for main, <script>, <body>, etc.).
- The function does not read any data (i.e., it may not have any prompt statements).
- The function does not print any data (i.e., it may not have document.writeln or alert).
- The function must work for any arrays (e.g., any number of rows and columns).

Problem 7, JavaScript (Randomize) (8 points)

The function `getRandomValue` has the prototype: **`function getRandomValue(lowerLimit, upperLimit)`**

The function returns a random integer value between `lowerLimit` (inclusive) and `upperLimit` (inclusive). For example, **`getRandomValue(5, 11)`** may return 5, 8, 11 or any other value between 5 and 11.

Assume that the **`getRandomValue`** function has already been written. Using the **`getRandomValue`** function, implement a function called **`randomize`** that has the following prototype:

`function randomize(dataArray);`

This function will randomize the elements of the **array** represented by **`dataArray`**. Notice that `dataArray` could be an array of any kind of elements (e.g., integers, strings, etc.).

Problem 8, JavaScript (Name Generation) (12 points)

Implement a function called **getArrayWithNames** that has the following prototype:

function getArrayWithNames(commonName, startNumber, endNumber)

The function returns an array with file names where all have the same common/base name, and use numbers in the range defined by startNumber and endNumber. For example, calling the method **getArrayWithNames("data", 2, 4)** returns an array with the values: **"data2"**, **"data3"**, **"data4"**. Remember that your function must work for values other than "data" and the numbers 2 and 4. If startNumber is greater than endNumber your program will use 1 as the start and end number.

Problem 9, JavaScript (Data Validation) (10 points)

Implement a function called **isBlank** that has as prototype **function isBlank(data)**. The function will return true if the string provided via the **data** parameter is blank, and false otherwise. We consider a string blank if the string has no characters at all ("") or if the only characters it has are blank characters. A blank character is represented by " followed by a space followed by a " (e.g. " "). Remember that you can use the **charAt()** function to determine the character at a particular index position. For example, if x = "hello", x.charAt(0) has the value "h". Below are some examples of calling the function you need to write.

isBlank("abc") → false
isBlank(" ") → true

isBlank("") → true
isBlank(" d ") → false

EXTRA SPACE IN CASE YOU NEED IT