University of Maryland College Park
Department of Computer Science
CMSC335 Fall 2022
Exam #1

FIRSTNAME, LASTNAME (PRINT IN UPPERCASE):

STUDENT ID (e.g., 123456789):

Instructions

- This exam is a closed-book, closed-notes exam with a duration of 75 minutes and 200 total points.
- You may lose credit if you do not follow the instructions below.
- **At this point, you must write your name and id at the top of this page and add your directory id (e.g., terps) at the end of odd-numbered pages.**
- Please use a pencil to answer the exam.
- **Do not remove the exam's staple or exam pages. Also, do not bend any of the pages, which will interfere with the scanning process.**
- Provide answers in the rectangular areas. If you continue a problem on another page(s), make a note. **For multiple-choice questions, please fill in the bubble (do not circle).**
- **Your code must be efficient and as short as possible.**
- For multiple-choice questions, you can assume only one answer unless stated otherwise.
- You don't need to use meaningful variable names; however, we expect good indentation.
- **You must stop writing once the time is up, otherwise you will lose significant credit.**

Grader Use Only

Problem #1 (Multiple-Choice)	63	
Problem #2 (CSS)	14	
Problem #3 (HTML)	30	
Problem #4 (Form)	26	
Problem #5 (JavaScript Code)	27	
Problem #6 (JavaScript Code)	40	
Total	200	

Problem #1 (Miscellaneous)

1. (3 pts) What is the value printed by the following function?

```
function task() {
   let age;

   document.writeln(age);
}
```

   a. undefined
   b. 0
   c. Empty string.
   d. A trash value (any random value).

2. (3 pts) In a web server, **localhost** corresponds to the IP address:

   a. 127.127.127.1
   b. 127.0.0.1
   c. 0.0.0.127
   d. None of the above.

3. (3 pts) In the Apache server provided by XAMPP, documents we want the web server to deliver to requests are in the directory:

   a. htdocs
   b. localhost
   c. index.html
   d. None of the above.

4. (3 pts) Server-side includes allow us to:
   a. Include the same HTML contents in several files.
   b. To combine several web server requests into one.
   c. Execute code in the browser.
   d. None of the above.

5. (3 pts) From what we discussed in the lecture, if we can see a page when we specify an URL that ends with the name of a directory (e.g., https://www.cs.umd.edu/class/fall2022/cmsc335 where cmsc335 is a directory), then:

   a. The directory has a text file called README.txt.
   b. The directory has a file called htdocs.
   c. The directory has a file called index.html or index.shtml.
   d. None of the above.

6. (3 pts) An HTTP **post** request:

   a. Can be bookmarked in the browser.
   b. Includes parameters in the URL.
   c. Usually does not modify the state of the server.
   d. None of the above.

7. (3 pts) The action attribute of the <form> tag is used to specify:

   a. The destination (e.g., a script) that will process the data.
   b. Whether HTTP get or HTTP post will be used.
   c. Whether encryption will be used.
   d. None of the above.

8. (3 pts) The DOM (Document Object Model) represents:

   a. An HTML document.
   b. The result of processing a CSS file.
   c. The first created object when we execute alert().
   d. None of the above.

9. (3 pts) The output of the following code fragment is:

```
let answer = ("20" === 20);
 document.writeln(answer);
```

   a. true
   b. false
   c. undefined
   d. null

10. (3 pts) The **prompt()** function returns:

   a. A number if one was entered by the user and a string otherwise.
   b. A string no matter what the user entered.
   c. null if the user enters a floating-point number.
   d. undefined if the user enters a floating-point number.

11. (3 pts) The output of the following code fragment is:

```
let a = [10, 20];
a.length = 4;
document.writeln(a[2]);
```

   a. true
   b. false
   c. undefined
   d. null

12. (3 pts) The output of the following code fragment is:

```
document.writeln(Number("87.76squareinches"));
```

   a. 87
   b. 87.76
   c. NaN
   d. null

13. (3 pts) The functions Window.NaN() and Number.NaN() always generate the same results.

   a. true
   b. False

14. (3 pts) Which of the following are considered falsy in JavaScript? Select all that apply.

   a. 1.5
   b. undefined
   c. NaN

15. (3 pts) Complete the following assignment so **x** is assigned a random floating-point value between 5 (inclusive) and 105 (exclusive).

```
let x =
```

16. (6 pts) Rewrite the following assignment using template literals.

```
let answer = "Value<strong>" + value + "</strong><br><em>Sqrt" + Math.sqrt(value) + "</em>";
```

```
let answer =
```

17. (12 pts) Complete the implementation of the **sortNumbers** function below. The function will sort the elements of the number **data** array in increasing order if the increasing parameter is true and in decreasing order otherwise. The following is an example of using the function.

Driver:	**Output:**
`let data = [10, 3, 89, 5];` `sortNumbers(data, true);` `document.writeln("First " + data.join() + " ");` `sortNumbers(data, false);` `document.writeln("Second " + data.join());`	`First 3,5,10,89` `Second 89,10,5,3`

```
function sortNumbers(data, increasing) {
```

Problem #2 (CSS)

1. (6 pts) Define a CSS rule that makes the size of paragraphs be 4.5 rem and the background color **blue**.

2. (4 pts) Define a CSS rule for the following HTML based on an **id selector** that makes the color of the **div** text yellow.

```
<div id="lot">
    Parking lot
</div>
```

3. (4 pts) Define a CSS rule for the following HTML based on a **class selector** that makes the font-size **3.5 em**.

```
<pre class="road">
   Road ahead
</pre>
```

Problem #3 (HTML)

1. (5 pts) Using the <img> tag, define an image entry where the image name is **car.png,** and the message "a car" will appear when the image cannot be displayed.

2. (15 pts) In the rectangular area below, write the HTML that goes inside of the <body></body> tags that will generate the following list. Notice that the second entry of the main list has a list. You may not type any numbers as part of your HTML.

- Study
- Clean
    1. Room
    2. Bath
- Dinner

DirectoryId (e.g., terps):

3. (10 pts) In the rectangular area below, write the HTML that goes inside of the <body></body> tags that will generate the following table. Do not use CSS. To create the border, use the table **border** attribute with a value of one. Do not use the tag to bold the header.

Task
P1

Problem #4 (Form)

Define the following form (just the form tags and its contents) that will call the script called **placeOrder.php** using the **post** method.

Firstname [ ] Place Order Reset

Problem #5 (JavaScript Code)

Implement the JavaScript function called **factorial** that computes the factorial of a number. If the parameter does not represent a number, the function will return NaN. To verify whether the parameter is a number, first try to turn the parameter into a number using Number(). You can assume that if function is called with a number, the number will be greater or equal to one. The following is an example of calling the function:

Driver:	**Output:**
`document.writeln(factorial(4) + " ");` `document.writeln(factorial("Rose") + " ");`	`24` `NaN`

DirectoryId (e.g., terps):

Problem #6 (JavaScript Code)

Implement a JavaScript function called **getFactorialTable** that returns a string with HTML representing a table of factorials. The function will take a parameter called **limit** representing up to what factorial will be part of the table. Use the **factorial** function you defined in the previous problem (even if you did not implement it) during the implementation of this function. You can assume the **limit** parameter will be a number greater than or equal to one. You can assume a border of 1 for the table (do not use CSS for this problem). For example, document.writeln(getFactorialTable(5)) will generate the following table:

1	1
2	2
3	6
4	24
5	120

LAST PAGE