Sample CS 142 Midterm Examination

Spring Quarter 2017

You have 1.5 hours (90 minutes) for this examination; the number of points for each question indicates roughly how many minutes you should spend on that question. Make sure you print your name and sign the Honor Code below. During the examination you may consult two double-sided pages of notes; all other sources of information, including laptops, cell phones, etc. are prohibited.

I acknowledge and accept the Stanford University Honor Code. I have neither given nor received aid in answering the questions on this examination.

____
(Signature)

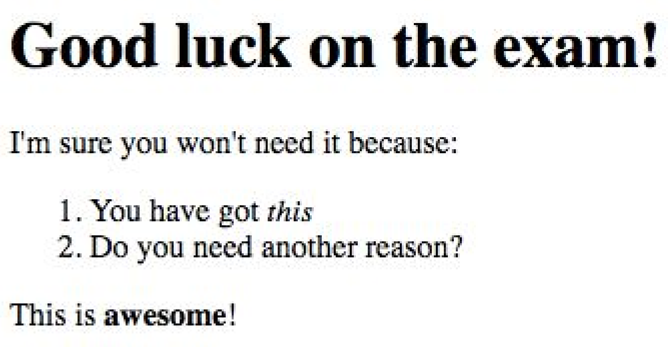
____
(Print your name, legibly!)

____________________________________@stanford.edu
(Stanford email account for grading database key)

Problem	#1	#2	#3	#4	#5	#6	#7	#8	#9	Total
Score										
Max	10	10	10	10	12	10	12	8	8	90

Problem #1 (10 points)

While on an archeological dig you come across a faded printout of a browser screen displaying an hypertext markup language (HTML) document from a time before cascading style sheets (CSS) were added to the browser. The printout looked like:



Good luck on the exam!

I'm sure you won't need it because:

1. You have got *this*
2. Do you need another reason?

This is **awesome**!

(a) The dig's specialist in ancient hypertext markup languages recognizes this as a document that uses six different HTML tags. List the six unique tags used in the body of the document (excluding the `<body>` tag).

Note: consider opening and closing tags (e.g. `<body></body>`) the same tag

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____

(b) With the introduction of CSS in browsers, some of the six HTML tags listed in part (a) would be better done using CSS. State which of the tags would be considered unnecessary with CSS available and describe why.

Problem #1 continued on next page….

…. continued from previous page.

(c) Around the campfire at the archeological dig the ancient HTML specialist was recounting the disagreements that led to the markup language splitting into the HTML and the XHTML camps. Describe the main disagreement that led to the two different markup language standards.

Problem #2 (10 points)

For each of the blanks in the code below, write what would be printed by the code on the line. If nothing is printed, write **nothing**. Hint: `Object.prototype` can be viewed as the top most object in the JavaScript object prototype chain.

```
Object.prototype.bar = function() { console.log("bar"); };
function Apple() {};
Apple.wash = function() { console.log("wash"); };
Apple.prototype.throw = function() { console.log("throw"); };

var myApple = new Apple();

if (myApple.wash) myApple.wash();     (a)____________________________

if (myApple.throw) myApple.throw();   (b)____________________________

if (myApple.bar) myApple.bar();       (c)____________________________

var yourApple = new Apple();

yourApple.throw = function() { console.log("throw2"); };
yourApple.wash = function() { console.log("wash2"); };
Object.prototype.bar = function() { console.log("bar2"); };

if (yourApple.wash) yourApple.wash();   (d)____________________________

if (yourApple.throw) yourApple.throw();(e)____________________________

if (yourApple.bar) yourApple.bar();     (f)____________________________

console.log(yourApple.throw === myApple.throw); (g)__________________

console.log(yourApple.wash === myApple.wash);   (h)__________________

console.log(yourApple.bar === myApple.bar);     (i)__________________
```

Problem #3 (10 points)

(a) You would like to load the page **`junipers.html`** from **`www.trees.org`**, using HTTP. Additionally, you want to pass the parameter **`species`** to the server with a value of **`californica`**. Please construct the URL you would use to access this page, and **label all components of the URL**.

(b) Having navigated to the page above, you see it contains several links. For each of the links below, circle the correct option:

```
<a href="/junipers.html#~cultivation">Link 1</a>
```

The type of this link is:	Absolute	Relative	Other
This link contains invalid characters:	True	False	
This link causes a page reload:	True	False	

```
<a href="/spruces.html#classification&naming">Link 2</a>
```

The type of this link is:	Absolute	Relative	Other
This link contains invalid characters:	True	False	
This link causes a page reload:	True	False	

```
<a href="#juniper_berry">Link 3</a>
```

The type of this link is:	Absolute	Relative	Other
This link contains invalid characters:	True	False	
This link causes a page reload:	True	False	

Problem #4 (10 points)

Below you see an HTML document and a blank browser tab (on the following page). Fill in the blank browser tab to show everything you will see when this HTML page is opened in the browser tab. You do not have to worry about drawing exact dimensions - simply approximate height, width, margins, etc. in your sketch.

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website</title>
    <style type="text/css">
      .box {
        display: inline-block;
        height: 50px;
        width: 50px;
        border: 1px solid black;
        margin-left: 10px;
        margin-top: 10px;
      }

      #box2 {
        background-color: black;
      }

      .container {
        display: block;
      }
    </style>
  </head>
  <body>
    <div class="container">
      <h1>BIG TITLE</h1>
      <div class="box"></div>
      <div class="box" id="box2"></div>
    </div>
  </body>
</html>
```

See answer space on next page….

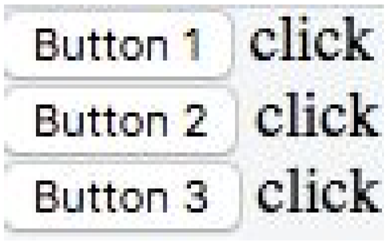
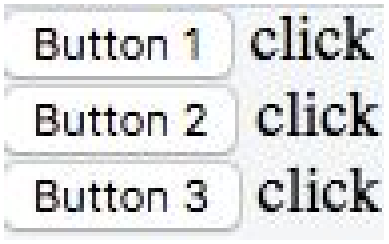
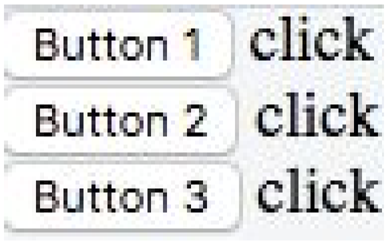
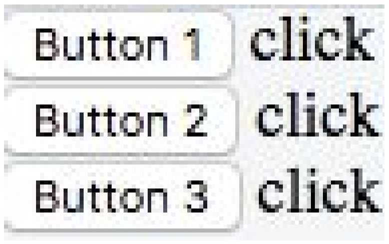
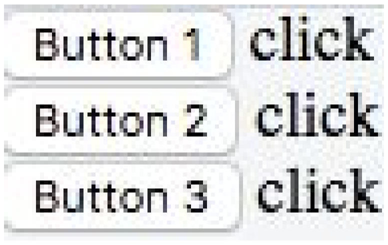
Continued from previous page….

Problem #5 (12 points)

The following Angular.js program displays three buttons (Button 1, Button 2, and Button 3) with the word "click" followed by a number. In the spaces provided below (15 total), write the number that will be displayed after the actions described. Hint: Recall that ng-repeat will create a new scope for each iteration.

```
<body ng-app="myApp" ng-init="count = 0">
  <button ng-click="count = count + 1">Button 1 </button>
  click {{count}}

  <div ng-repeat="button in [2,3]">
    <button ng-click="count = count + 1">Button {{button}}</button>
    click {{count}}
  </div>
</body>
```

(a) The program is first started in the browser: 	(d) After step (c), Button 1 is pushed 1 time: 
(b) After step (a), Button 1 is pushed 2 times: 	(e) After step (d), Button 2 is pushed 2 times: 
(c) After step (b), Button 3 is pushed 3 times: 	

Problem #6 (10 points)

The following JavaScript program prints some lines to the console log. In the places indicated below, describe what is printed to the log.

```
var gColor = 'blue';

function CrayonStore() {

  var color = 'red';

  this.getFunc = function() {
    this.color = 'purple';
    return function() {
      console.log(this && this.color);
      console.log(color);
      console.log(gColor);
    };
  };

  this.getFuncObject = function() {
    var colors = ['green', 'blue'];
    var retObj = {};
    for (var idx = 0; idx < colors.length; idx++) {
      retObj[colors[idx]] = function() { console.log(colors[idx]); };
    }
    return retObj;
  };
}

var crayonStore = new CrayonStore();
var func = crayonStore.getFunc();
func();  // (a) What is printed by this call?



gColor = 'green';
crayonStore.color = 'black';
func();  // (b) What is printed by this call?



var objectFunc = crayonStore.getFuncObject();
objectFunc.green();  // (c) What is printed by this call?


objectFunc.blue();  // (d) What is printed by this call?
```

Problem #7 (12 points)

(a) Using the tree structure of the DOM it is possible to navigate to an arbitrary DOM node by starting at the root of the tree and traversing the tree properties (e.g. `document.body.firstChild.nextSibling.firstChild`). Explain why this kind of navigation is not the preferred way of getting to DOM nodes.

(b) In the lecture notes we mentioned that for a DOM node the `offsetParent` is not necessarily the same as `parentNode`. Use a code fragment to describe something you could do in HTML and/or CSS that would cause these two properties to be different.

Problem 7 continued on next page ….

… continued from previous page.

(c) Why do you think the bubble phase of event listeners is more commonly used than the capture phase? Please specify the difference between the two phases in your answer.

Problem #8 (8 points)

(a) CSS supports a variety of different units to specify distances, including `px`, `in`, `pt`, and `em`. Which of these units would be appropriate for a web application supporting responsive web design? Justify your answer(s). (There may be more than one.)

(b) Responsive web application design makes use of CSS breakpoints to handle different sized displays. How might you decide the dimensions at which to place the breakpoints in your styling? Use specific examples to justify your answer.

Problem #9 (8 points)

(a) The browser's navigation bar (e.g. forward/back buttons, refresh button, bookmarks) can be a challenge for JavaScript frameworks like Angular.js. Explain the basic problem with it and describe the solution used to make it work with JavaScript frameworks.

(b) For each of the following items from web application development, which (if any) of the web view components (i.e. Model/View/Controller) would you expect to take the brunt of the challenges to implement the item. Justify your answer.

(i) Internationalization

(ii) Accessibility

(iii) End-to-End testing