FINAL EXAMINATION – DECEMBER 2002
CSC 105 – INTERACTIVE WEB DOCUMENTS
NICHOLAS R. HOWE

This is an open-book, open-notes exam.

All answers should be written in your exam booklet(s). Start with the questions that you know how to do, and try not to spend too long on any one question. Partial credit will be granted where appropriate. You will have two hours and twenty minutes. Good luck!

Cascading Style Sheets

1. (10 points) Write a CSS rule that would display all `<h1>` tags in centered, small-caps, aqua text, in Arial font. (Be sure to propose reasonable alternative fonts, including one for Macintosh users.)

2. (12 points) Imagine that a new CSS characteristic has been created, called `texture`. The possible values for `texture` are `smooth`, `bumpy`, `furry`, `ridged`, and `pitted`. Assume that the default `texture` is `smooth`, except for text within a new tag, `<f>`, which is `furry` by default. (The `<f>` tag is designed to work just like `<i>` and `<b>` do.) Given the CSS rules below, predict the displayed `texture` of the paragraphs 1-6.

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title></title>
<style type="text/css">
p.one {
  texture: bumpy;
}
#two {
  texture: pitted;
}
#three {
  texture: furry;
}
p em {
  texture: ridged;
}
</style>
</head>

<body>
<p><b>Paragraph 1<b></p>
<p class="one">Paragraph 2</p>
<p class="one" id="two">Paragraph 3</p>
<p class="one" id="three" style="texture: ridged">Paragraph 4</p>
<p class="one"><f>Paragraph 5</f></p>
<f><p class="one">Paragraph 6</p></f>
<p><em>Paragraph 5</em></p>
<em><p>Paragraph 6</p></em>
</body>
</html>
```

JavaScript

3. (12 points) Predict the contents of the variable array `arr` after the script below has run. Draw your answer inside a series of boxes representing the array, as we did in class.

```
var i
var arr = new Array
for (i = 0; i < 6; i++) {
  if (i%2 == 0) {
    arr[i] = "even"
  } else {
    arr[i] = "odd"
  }
  arr[i] = arr[i]+" ("+(i*i)+")"
}
```

4. (12 points) Consider the following script:

```
document.getElementById("div1").left = 10
document.getElementById("div1").top = 100-10
document.getElementById("div1").left = 20
document.getElementById("div1").top = 100-20
document.getElementById("div1").left = 30
document.getElementById("div1").top = 100-30
document.getElementById("div2").left = 15
document.getElementById("div2").top = 100-15
document.getElementById("div2").left = 25
document.getElementById("div2").top = 100-25
document.getElementById("div2").left = 35
document.getElementById("div2").top = 100-35
```

a. Simplify the script by writing two functions taking a single numeric input parameter. You will have one function for `div1`, and one for `div2`. Using these functions, replace the twelve lines above with six function calls.

b. Now write just one very general function that takes two input parameters. Replace the twelve lines above with six calls to your single function.

Document Object Models and Dynamic HTML

5. (16 points) Write lines of JavaScript to make the following dynamic changes, using each of the three document object models discussed in class (Netscape 4, Internet Explorer 4, W3C). You should write one line for each type of DOM, rather than using the functions developed in class for `dom.js`.

a. Change the `visibility` of `<div id="main">` from `visible` to `hidden`.

b. Change the `color` of `<p id="para1">` from `red` to `blue`.

6. (8 points) Specify the event trigger(s) you would use to create the following dynamic behaviors:

Example: For a rollover, you need the `onMouseOver` and `onMouseOut` triggers.

a. To reformat the page when the user adjusts the browser window dimensions.

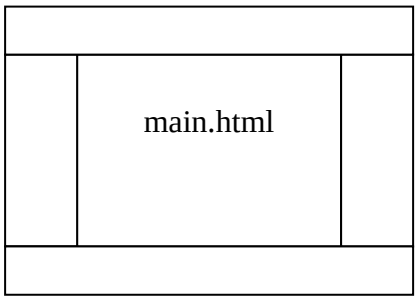
b. To stop an animation effect when the open browser window is covered by another window.

c. To check a form for errors before the results are sent to be processed.

d. To recompute the cost of an order when a user changes the number of items in a text box of a form.

Frames

7. (12 points) Write a piece of HTML that will create the set of frames pictured below. The center frame should always be 300 pixels tall and 500 pixels wide; the other frames should adjust themselves so that the center frame is in the middle of the window. (You don't need to write an entire HTML document; the portion from the first `<frameset>` to the last `</frameset>` will do.) The center frame should display the document `main.html`, while all the other frames should display `margin.html`. If viewed in a browser that doesn't understand frames, the user should see a link to `main.html`.



Forms

8. (12 points) Given the form created by the piece of HTML below, what would be the full url (including the assembled result string) that the browser would jump to when the form was submitted? Assume that none of the default values are changed before submission.

```
<form action="http://www.smith.edu/exam.html" method="get">
<input type="text" name="name" id="name" value="Mickey Mouse"></input></td>
<input type="checkbox" name="citizen" value="yes"></input>
<input type="radio" name="food" id="f1" value="cheese"
checked="checked"></input>
<input type="radio" name="food" id="f2" value="crackers"></input>
<input type="radio" name="food" id="f3" value="peanut butter"></input>
<input type="button" name="honk" value="beepbeep"></input>
<select name="menu" id="menu">
<option name="m1" id="m1" value="Huey">Huey</option>
<option name="m2" id="m2" value="Dewey" selected="selected">Dewey</option>
<option name="m3" id="m3" value="Louie">Louie</option>
</select>
<button type="submit" name="submit" value="submit">Submit</button>
</form>
```

State Maintenance

9. (10 points) Write a few paragraphs comparing the relative benefits and drawbacks of cookies vs. hidden frames for maintaining information as a user moves from page to page within a site. Be sure to cite specific advantages and disadvantages of each method.