**MIDTERM II**

1. http is the protocol. www.cs.dartmouth.edu is the host. ~fabio/teaching/cs4-winter07/ is the path and schedule.html is the file.

2. The cellphone or iPhone has a small screen. Hence, if we use the fixed layout, these sites will only show a small portion of their homepages.

3. <h1> music </h1>
   <img src = "music.jpg" </img>
   <ul>
   <li> Punk </li>
   <li> Funk </li>
   </ul>

4. p.r = {color:rgb(255,0,0);}
   p.b = {color:rgb(0,0,255);}
    change me to red 
    change me to blue 
    change me to red 
    Do not change 

5. The & sign will go wrong. We should write &
   'dancing '
   'salsa '
    is not closed
    is not closed

6.

| Day | Topic | Reading |
|---|---|---|
| Today | Midterm | |

7. An algorithm is an ordered collection of unambiguous and effectively computable operations that, when followed produces an observable result, and completes (halts) in a finite amount of time.
8. We should use binary search when the array is already sorted. The nice thing about binary search is that it saves a lot of computing time. It only needs roughly log n comparisons (compared to the possibly n comparisons in the sequential search).
9. The algorithm returns the number of elements in A greater than the variable v
10. The while loop may run former. (Observe carefully what happens if the "if" statement is true?)
11. The single equal sign means "assignment"; the double equal sign means "comparison" (usually used in the conditional statements). They are logically different, so we use different notation to make things clear.

12. The first line assigns a boolean value to the variable v1; while the second line assigns a string to the variable v2. The third line is wrong. We cannot "add" a boolean value to another Boolean value. However, the fourth line is correct. It is legitimate to concatenate two different strings by the "+" sign.
13. 
```
var dancing;
if (age > 60)
  dancing = "Not Really";
else if (age >35)
   dancing = "Maybe";
else
    dancing = "Surely";
```
14. 
```
for (i=0; i < A.length; i +=1) {
   if (A[i]%2 == 0){
      A[i] = A[i] +1;
   }
}
```

Round 1: ret = 0, counter =0, stop = false, A[counter]=12
Round 2: ret = 1, counter =1, stop = false, A[counter]=10
Round 3: ret = 1, counter = 2, stop =false, A[counter]=9
Round 4: ret = 2, counter = 3, stop = false, A[counter] = 6

This function counts how many numbers greater than v is a multiple of 3. It won't work if the given input is in the reverse order.