COSC 455
Final Review Sheet
Spring 2016

1.) For each of the following, indicate whether $f = O(g)$, $f = \Omega(g)$, or $f = \Theta(g)$:

	$f(n)$	$g(n)$
a.	$100n + \log n$	$n + (\log n)^2$
b.	$n^2 / \log n$	$n(\log n)^2$
c.	$n^{1/2}$	$5^{\log_2 n}$
d.	$n!$	2^n
e.	$(\log n)^{\log n}$	$2^{(\log_2 n)^2}$

2.) Solve the following recurrence relations and give a O bound for each:

a.) $T(n) = 7T(n/7) + n$
b.) $T(n) = 9T(n/3) + n^2$
c.) $T(n) = 8T(n/2) + n^3$

3.) Write the recurrence and solve the recurrence for the following function:

```
function f(n):
   if n>1:
      print("still going")
      f(n/2)
      f(n/2)
```

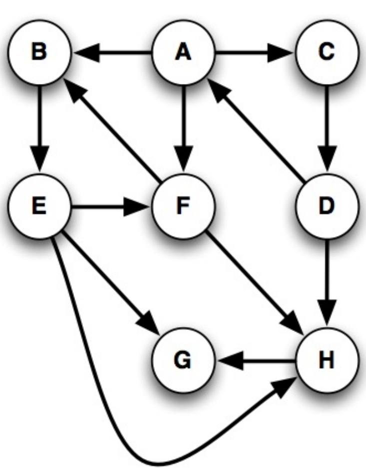
4.) Consider the standard algorithm for mergesort. Give the recurrence for mergesort and solve that recurrence.

5.) Suppose that you have k sorted arrays, each with n elements, and you want to combine them into a single sorted array of kn elements. Create an algorithm that is more efficient than $O(k^2 n)$.

6.) The Dutch Flag Problem is to arrange an array of the characters $\{R, W, B\}$ such that all of the R's come first, then the W's, then the B's. Give a linear-time, in-place algorithm to do this arrangement.

7.) Show that binary search is $O(\log n)$.

8.) Consider the following graph:



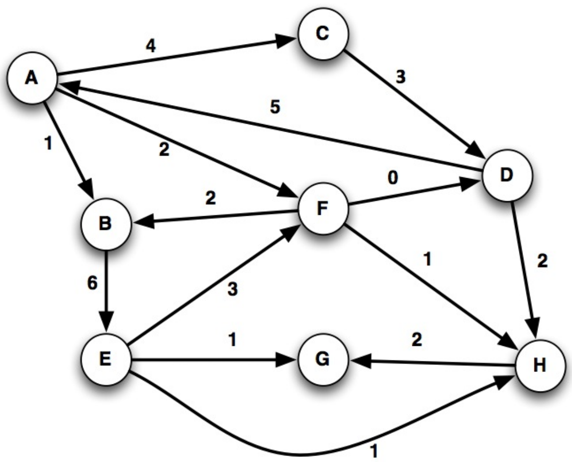
Construct the DFS tree. Add in and clearly identify forward edges, back edges, and cross edges. Assume that if there is a choice, we will expand out in alphabetical order.

9.) Run the strongly connected components algorithm on the graph in #8. Clearly show all steps.

10.) A graph is said to be bipartite if all of its vertices can be partitioned into two disjoint subsets X and Y such that edges only connect a vertex from X to a vertex in Y. There are no edges within the subsets. Using DFS, can I detect if a graph is bipartite? Can I do it with BFS?

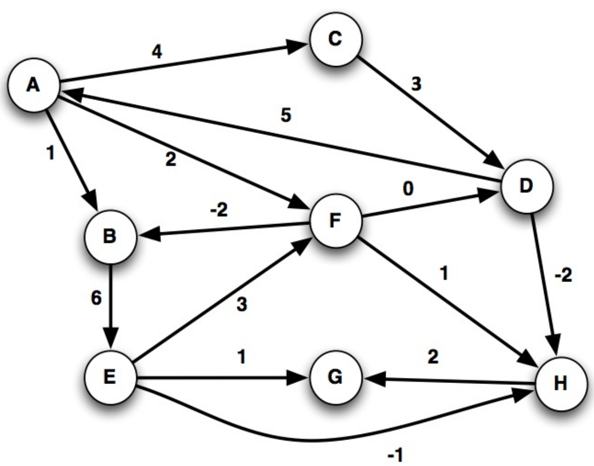
11.) Give a linear time algorithm which takes as input a directed graph, and determines whether or not there exists a vertex $s \in V$ from which all other vertices are reachable.

12.) Consider the following graph:



Run Dijkstra's algorithm on this graph, with A as the source. Show all steps.

13.) Consider the following graph:



Run the Bellman-Ford algorithm on this graph, with A as the source. Show all steps.

14.) How can we use the Bellman-Ford algorithm to detect if there exists a negative cost cycle in my graph?

15.) Consider the problem of determining all-pairs shortest path. For this problem, after your algorithm executes, each vertex s should have an array [1...n] that holds a list of the shortest path distances from s to each other vertex. Basically, determine the shortest path between all pairs of distinct vertices i and j. Give an algorithm that solves this problem. What is the time complexity of your algorithm?

16.) Given the graph in #13, run Kruskal's algorithm on that graph. Show all steps.

17.) Given the graph in #13, run Prim's algorithm on that graph. Show all steps.

18.) Consider encoding a data string containing $\Gamma = \{A, B, C, D, E, F, G\}$ with frequency (in percentage) $F = \{40, 10, 2, 20, 3, 5, 15\}$. Use Huffman prefix-free encoding and list the bit string symbols for each symbol in my alphabet. Draw the binary tree representation of the encoding.

19.) Fully describe and explain the DFS algorithm, including correctness.

20.) Fully describe and explain Dijkstra's algorithm, including correctness.

21.) Given the strings

```
x = "SNOWY", y="SUNNY"
```

show the table that is created by running the edit distance algorithm introduced in class.

22.) Given the items:

- A, value = 12, size = 5
- B, value = 6, size = 3
- C, value = 5, size = 2
- D, value = 2, size = 1

Determine what items are chosen given a knapsack of size 21 using the knapsack without repetition algorithm.

23.) Define dynamic programming.

24.) Design a dynamic programming solution (via pseudocode) for the maximum contiguous subsequence sum problem. You should take in an array of integers, S, and output a list containing consecutive elements of S that has the maximum sum.

25.) Define linear programming.

26.) (Hillier and Liberman) The WorldLight Company produces two light fixtures (products 1 & 2) that require both metal frame parts and electrical components. Management wants to determine how many units of each product to produce so as to maximize profit. For each unit of product 1, 1 unit of frame parts and 2 units of electrical components are required. For each unit of product 2, 3 units of frame parts and 2 units of electrical components are required. The company has 200 units of frame parts, and 300 units of electrical components. Each unit of product 1 gives a profit of $1, and each unit of product 2 gives a profit of $2. Any excess over 60 units of product 2 brings no profit, so such an excess has been ruled out. Formulate this problem as a linear programming problem.

27.) Define NP. Define NP-complete. What is a general-purpose approach to solving a NP problem?

28.) Show that vertex cover is NP-complete. A vertex cover is defined as a set $S \subseteq V$ such that every edge is incident (next to) a vertex in S. The problem is formulated as:

INSTANCE: Graph G, positive integer k
QUESTION: Does G have a vertex cover of size at most k?

29.) Fully describe the quantum portion of Shor's algorithm.

30.) How do we limit error in Monte Carlo algorithms?

31.) Describe how the Cole-Vishkin algorithm works. This is not an exercise of reproducing the algorithm, this is an exercise on being able to translate an algorithm into natural language.

32.) What NP-complete problems can Super Mario Bros. and sudoku reduce to?