CMPS 101
Winter 2017
Midterm Exam 2 Solutions

1. (20 Points) Use the Master Theorem to find tight asymptotic bounds for the following recurrences.

   a. (10 Points) $T(n) = 8T(n/4) + n\sqrt{n}$

   Solution:
   We compare $n\sqrt{n} = n^{3/2}$ to $n^{\log_4(8)}$. Since $8 = 4^{3/2} \Rightarrow \log_4(8) = 3/2 \Rightarrow n^{\log_4(8)} = n^{3/2}$, the two functions are asymptotically equivalent. Case 2 of the Master Theorem gives $T(n) = \Theta\big(n^{3/2}\log(n)\big)$.
   ■

   b. (10 Points) $T(n) = 3T(n/2) + \log(n!)$

   Solution:
   First, we simplify the recurrence: $T(n) = 3T(n/2) + n\log(n)$. (Recall we showed using Stirling's formula that $\log(n!) = \Theta(n\log(n))$.) Now we compare $n\log(n)$ to $n^{\log_2(3)}$.

   Let $\epsilon = \frac{1}{2}(\log_2(3) - 1)$. Then $3 > 2 \Rightarrow \log_2(3) > 1$, so $\epsilon > 0$. Also $1 + \epsilon = \log_2(3) - \epsilon$, from which it follows that

$$\lim_{n\to\infty}\left(\frac{n\log(n)}{n^{\log_2(3)-\epsilon}}\right) = \lim_{n\to\infty}\left(\frac{n\log(n)}{n^{1+\epsilon}}\right) = \lim_{n\to\infty}\left(\frac{\log(n)}{n^{\epsilon}}\right) = 0$$

   Therefore $n\log(n) = o\big(n^{\log_2(3)-\epsilon}\big)$, whence $n\log(n) = O\big(n^{\log_2(3)-\epsilon}\big)$. Case 1 of the Master Theorem says $T(n) = \theta\big(n^{\log_2(3)}\big)$. ■

2. (20 Points) Let T be a tree with n vertices and m edges. Use induction on m to prove that $m = n - 1$.
   Proof:
   I. Suppose $m = 0$. Since T is connected it can have only one vertex, hence $n = 1$. Thus $m = 0 = n - 1$, showing that the base case is satisfied.
   II. Let $m > 0$, and assume any tree T' with fewer than m edges satisfies $|E(T')| = |V(T')| - 1$. We must show that $m = n - 1$. Pick an edge e in T and remove it. The resulting graph $T - e$ consists of two trees T_1 and T_2, each with fewer than m edges. Suppose T_i has m_i edges and n_i vertices (for $i = 1, 2$). The induction hypothesis yields $m_i = n_i - 1$ (for $i = 1, 2$). Observe also that $n_1 + n_2 = n$ since no vertices were removed. Therefore

$$m = m_1 + m_2 + 1 = (n_1 - 1) + (n_2 - 1) + 1 = n_1 + n_2 - 1 = n - 1$$

   The result holds for all trees by the second principle of mathematical induction. ■

3. (20 Points) Let G be an acyclic graph with n vertices, m edges, and k connected components. Prove that $m = n - k$. (Hint: use the result of problem 2.)

Proof:

Let $T_1, T_2, \ldots, T_k$ be the connected components of G, and suppose T_i has n_i vertices and m_i edges respectively ($1 \le i \le k$). Since each T_i is necessarily a tree, we can apply the result of the previous problem to get $m_i = n_i - 1$ ($1 \le i \le k$). Then

$$m = \sum_{i=1}^{k} m_i = \sum_{i=1}^{k} (n_i - 1) = \sum_{i=1}^{k} n_i - \sum_{i=1}^{k} 1 = n - k,$$

and $m = n - k$ as required. ///

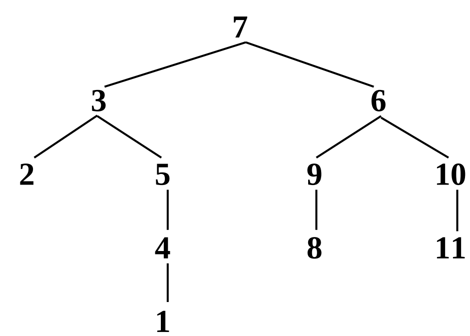
4. (20 Points) Run BFS on the digraph pictured on the back page of this exam, with source vertex $s = 7$. Process adjacency lists (the for loop on lines 12-17 of BFS) in increasing order by vertex label.

a. (10 Points) Fill in the following table, determine the order in which vertices enter the Queue, and draw the BFS Tree.

	Adjacency List	**Color**	**Distance**	**Parent**
1	2	black	4	4
2	5	black	2	3
3	2 5	black	1	7
4	1	black	3	5
5	4	black	2	3
6	3 5 9 10	black	1	7
7	3 6	black	0	NIL
8	4	black	3	9
9	4 5 8	black	2	6
10	9 11	black	2	6
11	7	black	3	10

Queue	7	3	6	2	5	9	10	4	8	11	1

BFS Tree:



b. (10 Points) Determine the shortest 7-1 path found by BFS. Find *all other* shortest 7-1 paths.

Shortest 7-1 path found by BFS: 7 3 5 4 1

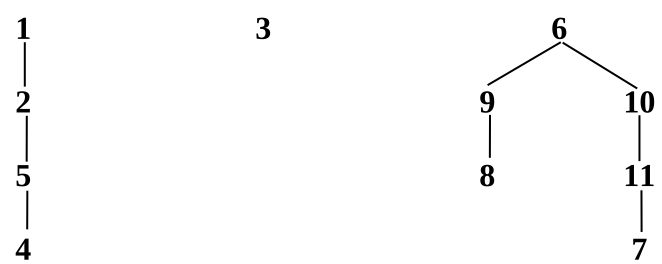
Two other shortest 7-1 paths: 7 6 5 4 1 and 7 6 9 4 1

5. (20 Points) Run DFS on the digraph pictured on the back page of this exam. Execute the main loop of DFS (lines 5-7) and process adjacency lists (the for loop on lines 3-6 of Visit) in increasing order by vertex label.

a. (10 Points) Fill in the following table, and draw the DFS Forrest.

	Adjacency List	**Discover**	**Finish**	**Parent**
1	2	1	8	NIL
2	5	2	7	1
3	2 5	9	10	NIL
4	1	4	5	5
5	4	3	6	2
6	3 5 9 10	11	22	NIL
7	3 6	18	19	11
8	4	13	14	9
9	4 5 8	12	15	6
10	9 11	16	21	6
11	7	17	20	10

DFS Forest:

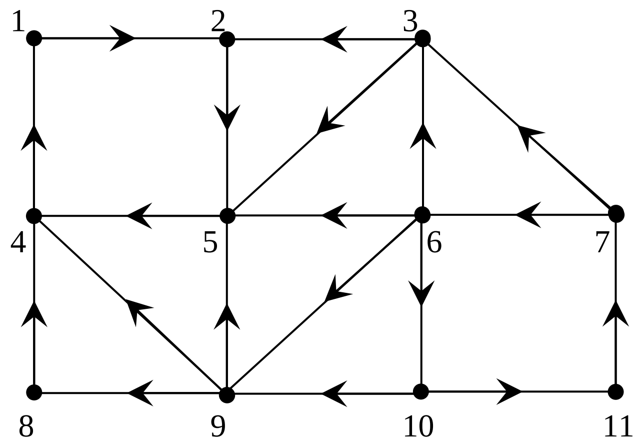


b. (10 Points) Classify all edges as tree, back, forward, or cross. Find all directed cycles in the digraph.

Tree	(1, 2), (2, 5), (5, 4), (6, 9), (9, 8), (6, 10), (10, 11), (11, 7)
Back	(4, 1), (7, 6)
Forward	
Cross	(3, 2), (3, 5), (6, 3), (6, 5), (7, 3), (8, 4), (9, 4), (9, 5), (10, 9)

Directed Cycles: 1 2 5 4 1 and 6 10 11 7 6

Problems 4 and 5 both refer to the following digraph:



The following algorithms are included for reference.

BFS(G, s)

```
for all x ∈ V(G) − {s}
    color[x] = white
    d[x] = ∞
    p[x] = NIL
color[s] = gray
d[s] = 0
p[s] = NIL
Q = ∅
Enqueue(Q, s)
while Q ≠ ∅
    x = Dequeue(Q)
    for all y ∈ adj[x]
        if color[y] == white
            color[y] = gray
            d[y] = d[x]+1
            p[y] = x
            Enqueue(Q, y)
    color[x] = black
```

DFS(G)

```
for all x ∈ V(G)
    color[x] = white
    p[x] = NIL
time = 0
for all x ∈ V(G)
    if color[x] == white
        Visit(x)
```

Visit(x)

```
color [x] = gray
d[x] = (++time)
for all y ∈ adj[x]
    if color[y] == white
        p[y] = x
        Visit(y)
color[x] = black
f[x] = (++time)
```