Fall 2009. Final Exam. Data Structures and Algorithms

Name:
Section:
Email id:

21st December, 2009 (Winter Solstice)

This is an open book, open notebook exam.
Answer all twelve questions. Each Question is worth 10 points. You have 180 minutes to complete the exam.
Happy Holidays and Have a Great New Year.

1. **NP-complete**

(a) Do Problem 8.13 c (DG)(Prove that the problem is NP-complete)
MAX SAT Given a CNF formula and an integer g, find a truth assignment that satisfies at least g clauses.

(b) Prove that clique problem can be solved in polynomial time for regular graphs of degree 3.

1.(a) ~~CNF~~ MAX SAT ~~is~~ is in NP.

guess an assignment
and see if at least g clauses are satisfied.

CNF SAT is polynomial time transformable to MAX SAT

Given: CNF SAT Problem
? : Does $\exists$ an assignment to variables that satisfies all clauses (Let number of clauses be t)

Instance for MAX SAT:
Given: P, g ($g = t$)
? : Does $\exists$ an assignment which satisfies g clauses

Assignment satisfies g clauses in MAX SAT iff the given CNF is satisfiable.

(b) Take subsets of 4 vertices and check ~~[illegible]~~ whether the given 4 vertices form a complete graph. report clique of size 4.
else
Take subsets of 3 vertices and check whether the given 3 vertices form a complete graph. report clique of size 3.
else clique is of size 4.

Complexity is $O(n^4)$

2. Local Search

Problem 9.6 (DG)

In the Minimum Steiner Tree Problem, the input consists of: a graph $G = (V, E)$ with distances d_{uv} between all pairs of nodes (distances satisfy metric property); and a distinguishable set of terminal nodes $V' \subset V$. The goal is to find an efficient ratio-2 approximation algorithm for finding a minimum Steiner tree that includes vertices V'. This tree may or may not include vertices $V - V'$. (Hint: Construct all pairs of shortest distances between vertices in V' and what can you say about these distances? Then construct a Minimum Spanning Tree and select edges from the original graph).

1. Find a minimum spanning tree T' from the induced subgraph of on V'!

2. claim the MST is at most twice the cost of Steiner tree.

Proof: Let T be a steiner tree.

Sketch:

v_1' (a) v_2' (a) (b) v_5' (b) v_6'
(b) (c) v_4' (c) v_3'
(c) (b) (a) v_1'

$c(v_1' v_2' v_5' v_6' v_4' v_3')$
$<$ 2 cost of Steiner tree

Do an Eulerian path traversal of this graph and consider only the V' vertices. the cost of the MST T' is at most twice cost of T

3. **Graphs/DFS/Path**

Consider the following graph with 10 vertices and 15 edges.

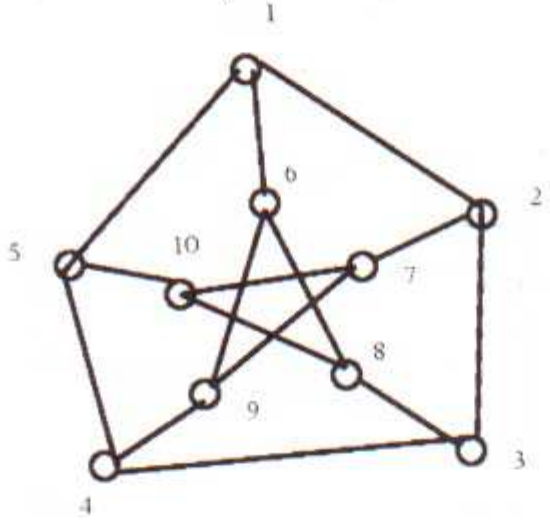
[**10 points**]

(a) Draw a BFS tree starting at vertex 1.

(b) Draw a DFS tree starting at vertex 1.

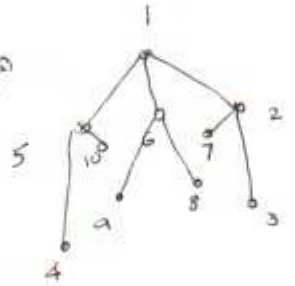
(c) What is the longest path from vertex 1 and vertex 10.

(d) Draw two vertex paths from vertex 1 to vertex 2.

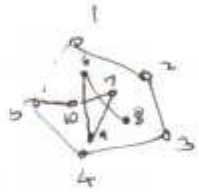


(c) 1-6-9-7-2-3-
4-5-10

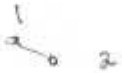
(a)

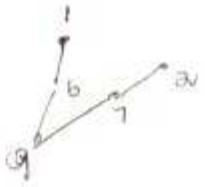


(b)



(d)

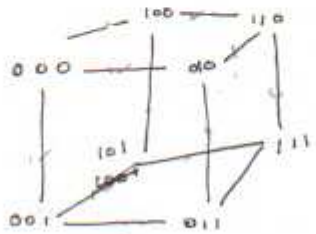




1. **Shortest Path Algorithms/Spanning Tree** Consider 8 of 3 bit binary numbers from 000 to 111. Treat them as vertices of an undirected graph. Two vertices are adjacent if they differ in exactly in one bit position. For example vertex labeled 001 is adjacent to 101, 011, 000.

   (a) Draw a graph of this 8 vertex undirected graph.

   (b) Draw a cycle of length 8 from vertex labeled 000 to vertex labeled 000.



000, 010, 011, 001, 101, 111, 110, 100, 000

5. **Linear Programming and Matching**

(a) Do problem 7.3 (just the LP formulation): A cargo plane can carry a maximum weight of 100 tons and maximum volume of 60 cubic meters. There are three materials to be transported, and the cargo company may choose to carry any element of each, up to the maximum available limits given below.

- Material 1 has density 2 tons/cubic meter, maximum available amount is 40 cubic meter and revenue is \$1000 per cubic meter.
- Material 2 has density 1 ton/cubic meter, maximum available amount is 30 cubic meter and revenue is \$1200 per cubic meter.
- Material 3 has density 3 ton/cubic meter, maximum available amount is 20 cubic meter and revenue is \$12000 per cubic meter.

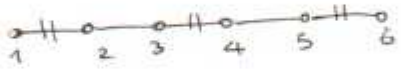
Write a linear program (just formulation) that optimizes revenue within the constraints. (No solution)

Let x_i be the cubic meter for material i

$$\max \quad 1000x_1 + 1200x_2 + 12000x_3$$

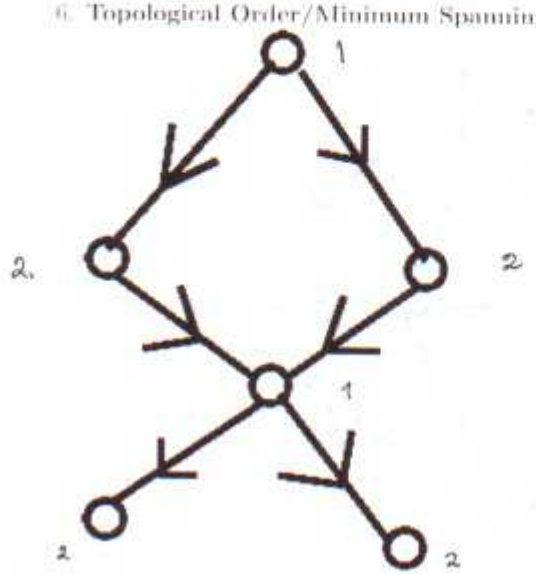
$$\text{s.t.} \quad 2x_1 + x_2 + 3x_3 \le 100$$
$$x_1 + x_2 + x_3 \le 60$$
$$x_1 \le 40$$
$$x_2 \le 30$$
$$x_3 \le 20$$
$$x_1, x_2, x_3 \ge 0$$

(b) Draw a tree with 6 vertices and has a max matching with three edges.



$\{\{1,2\}, \{3,4\}, \{5,6\}\}$ is a matching

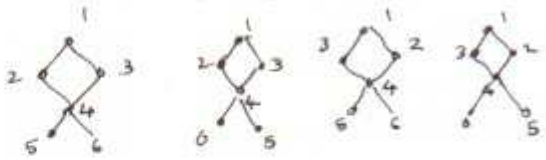
6. Topological Order/Minimum Spanning Tree



(a) Consider the directed acyclic graph G in the above figure. How many topological orderings does it have? (What are all the possible labeling of the vertices that preserve topological order.)

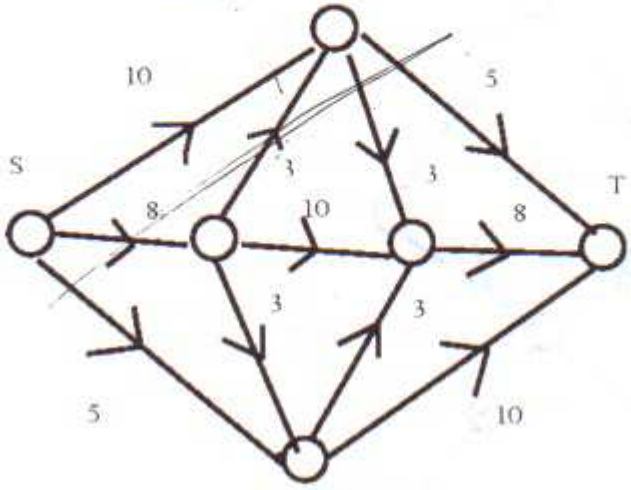
(b) Describe an algorithm to find the largest spanning tree in a given undirected weighted graph.

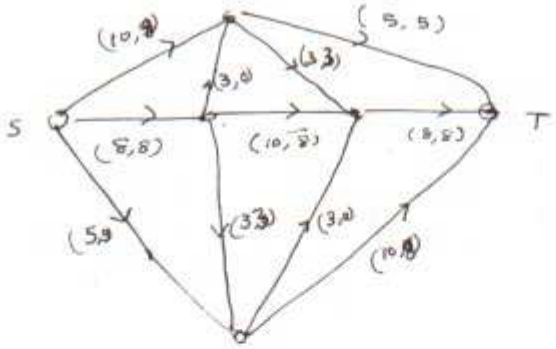
(a) 4 possibilities



(b) In Kruskal's algorithm sort the edges in descending order of weights. Select the largest weight edge and see if it can be included in the max. Sp. Tree.

7. **Network Flows** Consider the following network (the numbers are edge capacities). What is the maximum flow from S to T and what is the minimum cut

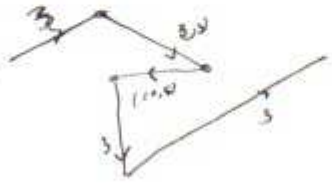




Max F
21

min cut is 21

+



8. **Dynamic Programming** Given two strings $x = x_1x_2\cdots x_n$ and $y = y_1y_2\cdots y_m$, we wish to find the length of their largest common substring, that is, the largest k for which there are indices i and j with $x_ix_{i+1}\cdots x_{i+k-1} = y_jy_{j+1}\cdots y_{j+k-1}$. Show how to do this in time $O(mn)$.

~~$F(x_1 \cdots x_i, y_1 \cdots y_j)$~~

~~$= F(x_1 \cdots x_{i-1}$~~

$S(i,j)$ = longest common substring of ~~ends~~ in $x[i]$ and $y[j]$
$x[1 \cdots i]$ and $y[1 \cdots j]$

intended in $\max_{i,j} S[i,j]$

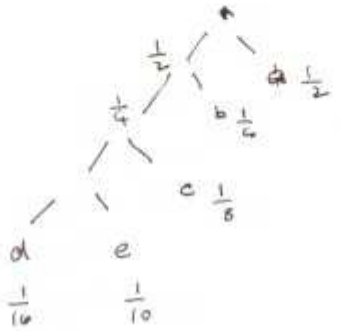
$$S(i,j) = \begin{cases} S(i-1, j-1) + 1 & \text{if } x[i] = y[j] \\ 0 & \text{otherwise.} \end{cases}$$

```
for i = 0 to n        s[i,0] = 0
for j = 0 to m        s[0,j] = 0;
    for i = 0 to n
        for j = 0 to m
            if x[i] == y[j]
                s[i,j] = s[i-1, j-1]+1
            else
                s[i,j] = 0;
return max_{i,j} s[i,j].
```

9. **Huffman** Suppose that the symbols a, b, c, d, e occur with frequencies of 1/2, 1/4, 1/8, 1/16, 1/16 respectively.

a) What is the Huffman encoding of this alphabet

b) If this encoding is applied to a file consisting of 1,000,000 characters with the given frequencies, what is the length of the encoded file in bits?



$a = 1$, $b = 01$, $c = 001$, $d = 0000$, $e = 0001$

length $1{,}000{,}000 \left(\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{16} \cdot 4 + \frac{1}{16} \cdot 4 \right)$

$1{,}000{,}000 \left(1 + \frac{3}{8} + \frac{1}{2} \right)$

$1{,}000{,}000 \left(\frac{15}{8} \right)$

$\frac{15{,}000{,}000}{8} = \underline{1{,}875{,}000}$

10. **Extended Euclidean Algorithm, Greedy**

(a) Give a linear time algorithm that takes input as a tree T and determines a two coloring of the vertices, such that each vertex is colored with one of two colors and two adjacent vertices are not colored with same color.

DFS on T.

Color the node as DFS Number of the node mod 2.

(b) Solve the equation for integers x and y such that $13 \times x + 8 \times y = 1$ $\gcd(13,8) = 1$.

Other Solution

$1 = 1 + 0$
$= 1 + (2 - 2 \cdot 1)$
$= 2 - 1 \cdot 1$
$= 2 - (3 - 1 \cdot 2)$
$= 2 \cdot 2 - 3$
$= 2(5 - 1 \cdot 3) - 3$
$= 2 \cdot 5 - 3 \cdot 3$
$= 2 \cdot 5 - 3(8 - 1 \cdot 5)$
$= 5 \cdot 5 - 3 \cdot 8$
$= 5(13 - 1 \cdot 8) - 3 \cdot 8$
$= 5 \cdot 13 - 8 \cdot 8$

$13x + 8y = 1$

$-11 \cdot 13 + 18 \cdot 8 = 1$

$13 = 1 \cdot 8 + 5$
$8 = 1 \cdot 5 + 3$
$5 = 1 \cdot 3 + 2$
$3 = 1 \cdot 2 + 1$
$2 = 2 \cdot 1 + 0$
$1 = 1 - 0$
$1 = 1 - (2 - 2 \cdot 1)$
$= -1 \cdot 2 + 3 \cdot 1$
$= -1 \cdot 2 + 3(3 - 1 \cdot 2)$
$= -4 \cdot 2 + 3 \cdot 3$

$3 \cdot 3 - 4(5 - 1 \cdot 3)$
$7 \cdot 3 - 4 \cdot 5$
$7 \cdot (8 - 1 \cdot 5) - 4 \cdot 5$
$7 \cdot 8 - 11 \cdot 5$
$7 \cdot 8 - 11(13 - 1 \cdot 8)$
$-11 \cdot 13 + 18 \cdot 8$

11. Algorithm Design

An array $A(1, \cdots, n)$ (random access) is said to have a majority element if more than half of its elements are the same. Given an array, design an efficient algorithm to tell whether there is a majority element, and, if so, find that element. The only question/operation that you are allowed to do is $A(i) = A(j)$? and this operation takes constant time. A pseudo code description will suffice. Analyze your algorithm.

```
potential max = -99999;
for i = 1 to n
{ if potential max = -99999
      potential max = A[i]
  else
      if potential max == A[i]
          count ++;
      else { count - - ;
             if count == 0
                potential max = -99999;
             }
  }
if potential max == -99999
    printf "there is no majority element".
else {count=0;
      for i = 1 to n
      ~~count the number of times potential max~~
      { if a[i] == potential max
            count++;
      }
      if count > n/2   print potential max
                  else  print no majority
                        element
```

$O(n)$

12. **Multiplication and Divide and Conquer**

(a) Describe an efficient algorithm to multiply two n digit complex integers (Hint: Use divide and conquer strategy). (A+IB) *(C+ID) = (AC-BD)+I(BC+AD)

$(A+B) * (C+D) = M_1$

$(A * C) = M_2$

$(B * D) = M_3$

$(M_2 - M_3) + I\,(M_1 - M_2 - M_3)$

(b) Find the value of T(8) and then Solve the following recurrence equation for general $n = 2^k$

$$T(n) = 3T(n/2)$$

and

$$T(1) = 1$$

$n = 2^k$

$T(2^k) = 3T(2^{k-1})$

$Q(k) = 3Q(k-1)$

$Q(1) = 1$

$\therefore Q(k) = 3^k \cdot 1.$

$T(2^k) = 3^k$

$T(n) = 3^{\log_2 n}$

$= n^{\log_2 3}$

$= O(n^{\log_2 3})$