The University of Sydney

Faculty of Engineering and Information Technologies
School of Information Technologies

COMP2007
Algorithms and Complexity

Midterm quiz
Semester 2, 2009
Time allowed: 50 minutes
Reading time: 10 minutes
CONFIDENTIAL

Enrolment: (circle one)	COMP2007 COMP2907
Last/Family name:	
First/Given name(s):	
SID:	
Signature :	

INSRTUCTIONS
Write answers in the spaces provided in this examination paper, using blue or black pen. Only one A4 sheet double sided of notes is permitted (that's 2 sides total, handwritten or computer printed). There is a total of 40 marks. Answer all questions.
Materials provided: None
Materials allowed: One A4 sheet of paper (2 pages) with notes, handwritten or printed

question	marks	total marks
1		10
2		10
3		20
Total		/4 0

Question 1 [10 marks]

This question has multiple true-or-false questions. For each question, mark in the given box one of the following two options TRUE, FALSE, as shown in the example below.

TRUE	Mergesort runs in time $O(n \log n)$ in the worst case
FALSE	Quicksort runs in time $O(n \log n)$ in the worst case

A correct TRUE/FALSE answer will add 1 mark, a wrong TRUE/FALSE answer will subtract 1 mark. Negative marks do not carry out of this question: the overall mark for question 1 will be between zero and 10 marks.

Mark the following statements TRUE or FALSE. Whenever a question refers to a graph, assume that n denotes the number of vertices and m the number of edges.

Ignore refs to
- min-cut, s-cut
- fbus, maxflow
- ford-fulkerson alg
(not covered yet)

1. T
2. F
3. F
4. F~
5. T

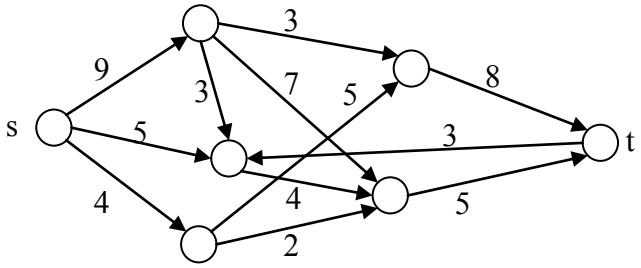
1 For any $n \geq 2$: $n^2 + n$ is $O(n^2)$

2 Prim's algorithm does not work for undirected graphs

3 If there are negative edge weights, Prim's algorithm cannot be used

4 The value of the maximum flow in a graph is equal to the capacity of the maximum cut in the graph

5 The shortest path from s to t in the graph below has cost 11.

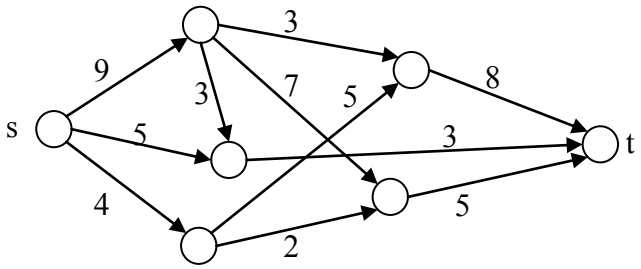


(Did Djkstra's on it)

6. T
7. ~

6 For any $n \geq 2$: n + $\log n$ is $O(n)$

7 The following graph has a minimum s-t cut of capacity 18



8. FALSE
9. FALSE
10. ~

8 All dynamic programming algorithms can only be implemented recursively

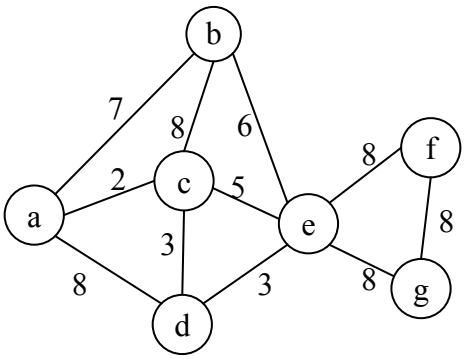
9 Dynamic programming always gives inefficient (exponential) algorithms because of the recursion

10 The running time of the Ford Fulkerson algorithm depends on the way we choose the augmenting paths on each round

Question 2 [10 marks]

This question has 2 parts, each one 5 marks.

a) Give the output (the minimum spanning tree) of Prim's algorithm on the graph below. Show all intermediate steps: for each step show which edge is added to the solution.
[5 marks]



AC
CD }
DE }
EB
EF }
EG }

//Tie breaks don't matter,
doesn't have to be connected.
//NOT DJKSTRA'S
[DJKSTRA's CONNECTED]

b) Give a different minimum spanning tree of the graph above (different from the solution described in your answer above). Would it be possible for any implementation of Prim's algorithm to return this second minimum spanning tree that you've found?
[5 Marks]

Question 3 [20 marks]

a) Recall that an undirected graph is called connected if there is a path between any pair of its vertices. Consider the problem of determining whether an undirected graph $G = (V, E)$ is connected or not. Design an algorithm that checks if a graph is connected based on Prim's algorithm for minimum spanning trees. Your answer should include a brief explanation how your algorithm works and what is its running time.
[8 marks]

Using Prim's algorithm from any one node, one should be able to reach all the nodes (V) of G. If this is not possible, then the graph is Not Connected. This is done by creating a MST of G using Prim's (by taking a subset of G, S, and expanding it one Vertex at a time with the minimum weight Edge from S). If at the end of this algorithm, the total V's don't match the nodes in G, then not connected.
Running time is the same amount of time taken to generate a MST via Prim's algorithm, this is O( ).
It is O( ) because in the worst case scenario, every node is connected to every other node, and each node must be compared with each other.

[———————————————————————————]

~ Cut Property

Set all Edges to 1.
Run Prim's Algorithm {
Every time you add a vertex to the tree,
counter++;
}
if counter==n;
return True
else; return False;

// So don't actually have to describe/explain Prim's Algorithm.
//The Above will do in the exam, do Not have to include Prim's.

//Always need to check for Cases.

Need to Explain WHY Works.
WHY
Explain Cases (One Sentence for Each Important Sentence).

Running Time; Same as Prim's (Prim + O(1))
Prims = O(m.logn) [WHERE M=|E|, N=|V|]
WHY

WE GET A SHEET TO BRING TO QUIZ

b) You are given an $n \times n$ matrix A, where each entry is either 1 or 0. We also know that in each row, all of the 1's occur before any 0's. Give an algorithm that counts the number of 1's in the matrix A in $O(nlog\ n)$ steps. Your answer should include an explanation how your algorithm works, and also an explanation of the running time of the algorithm. [12 marks]

Binary Search

See a 1 go Right, See a 0 go Left
Stop when .1.0. adjacent.

This is the end of the exam paper. You can use the rest of the page(s) for rough work