

CS570
Analysis of Algorithms
Fall 2015
Exam I

Name: _____

Student ID: _____

Email Address: _____

_____ **Check if DEN Student**

	Maximum	Received
Problem 1	20	
Problem 2	15	
Problem 3	12	
Problem 4	9	
Problem 5	12	
Problem 6	9	
Problem 7	10	
Problem 8	13	
Total	100	

Instructions:

1. This is a 2-hr exam. Closed book and notes
2. If a description to an algorithm or a proof is required please limit your description or proof to within 150 words, preferably not exceeding the space allotted for that question.
3. No space other than the pages in the exam booklet will be scanned for grading.
4. If you require an additional page for a question, you can use the extra page provided within this booklet. However please indicate clearly that you are continuing the solution on the additional page.

1) 20 pts

Mark the following statements as **TRUE** or **FALSE**. No need to provide any justification.

[**TRUE/FALSE**]

In a connected undirected graph, and using the **same** starting point, the depth of any DFS tree is at least as much as the depth of any BFS tree.

[**TRUE/FALSE**]

Algorithm A has a running time of $\Theta(n^2)$ and algorithm B has a running time of $\Theta(n \log n)$. From this we can conclude that A can never run faster than B on the same input set.

[**TRUE/FALSE**]

Let T be a complete binary tree with n nodes. Finding a path from the root of T to a given vertex $v \in T$ using breadth-first search takes $O(\log n)$ time.

[**TRUE/FALSE**]

Amortized cost of operations in a Fibonacci heap is at least as good as the worst case cost of those same operations in a binomial heap.

[**TRUE/FALSE**]

Master's Theorem can be used to calculate the running time of any recursive function.

[**TRUE/FALSE**]

Dijkstra's shortest path algorithm can be used to find shortest path in graphs with any edge weights.

[**TRUE/FALSE**]

Function $f(n) = 5n^2 4^n + 6n^4 3^n$ is $O(n^4 3^n)$.

[**TRUE/FALSE**]

Stable Matching: Suppose Jack prefers Rose to others, and Rose prefers Jack to others. The pair (Jack, Rose) exists in every stable matching.

[**TRUE/FALSE**]

A DFS tree is a spanning tree.

[**TRUE/FALSE**]

A binary max-heap can be built using an unsorted list of elements in $O(n)$ time

2) 15 pts

Suppose that you want to get from vertex s to vertex t in a connected undirected graph $G = (V; E)$ with positive edge costs, but you would like to stop by vertex u if it is possible to do so without increasing the length of your path by more than a factor of α .

- a- Describe an efficient algorithm in $O(|E| \log |V|)$ time that would determine an optimal s - t path given your preference for stopping at u along the way if doing so is not prohibitively costly. (In other words, your algorithm should either return the shortest path from s to t , or the shortest path from s to t containing u , depending on the situation.) If it helps, imagine that there are free burgers at u !
- b- Show that your algorithm runs in $O(|E| \log |V|)$.

3) 12 pts

Assume the coastline of the country Lyniera is an infinite straight line. There are islands off the coastline of Lyniera. In order to keep a close eye on these islands, the king of Lyniera decided to set up some radar bases along the coastline. Each radar base is a point on the coastline which can cover a circular area around itself with radius d . Let's use the x-axis in a coordinate system as the coastline (horizontal axis), with the sea on the upper side of the x-axis. Each island is a point in the sea with coordinates (x, y) . Design a greedy algorithm to find the minimum number of radar bases needed to cover all the islands and give their locations on the coastline. Prove the correctness of your algorithm. (Assume each island is within distance d of the coast.)

4) 9 pts

Solve the following recurrences using the Master Method. You do not need to justify your answers, but any justification that you provide will help when assigning partial credit.

i. $T(n) = 8T(n/2) + n \log n$

ii. $T(n) = 4T(n/2) + n^2 (\log n)^2$

iii. $T(n) = 2T(n/2) + n^3 (\log n)^3$

5) 12 pts

In the MERGE-SORT algorithm we merge two sorted lists into one sorted list in $O(n)$ time. Describe an $O(n \log k)$ -time algorithm to merge k sorted lists into one sorted list, where n is the total number of elements in all the input lists. Be sure to explain why your algorithm runs in $O(n \log k)$ -time.

6) 9 pts

Indicate for each pair of expressions (A,B) in the table below, whether A is O , Ω , or Θ of B (in other words, whether $A=O(B)$, $A=\Omega(B)$, or $A=\Theta(B)$). Assume that k and C are positive constants. You can mark each box with **Yes** or **No**. No justification needed.

(Note: log is base 2)

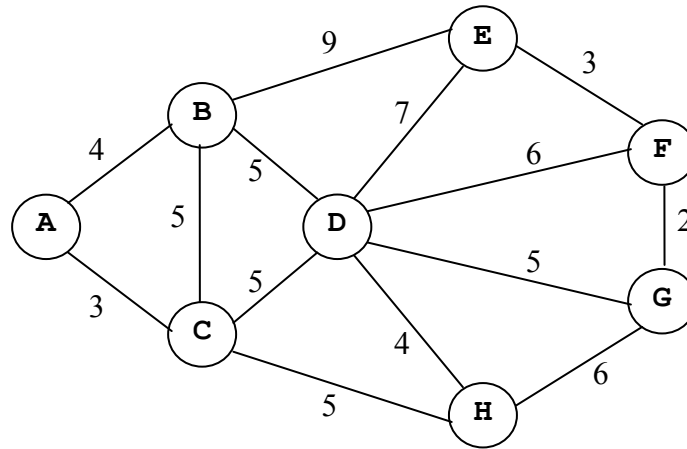
A	B	O	Ω	Θ
$n^3 + \log n + n^2$	Cn^3			
n^2	$Cn \cdot 2^{\log(n)}$			
$2^n \cdot 2^k$	$n^{(2k)}$			

7) 10 pts

Design an algorithm which, given an undirected graph $G = (V, E)$ and a particular edge $e \in E$ determines whether G has a cycle containing e . The running time should be bounded by $O(|V| + |E|)$. Explain why your algorithm runs in $O(|V| + |E|)$ time.

8) 13 pts

Given the undirected graph shown below:



- a- Use Prim's algorithm to obtain an MST of this graph. Use A as your starting point. Show the final MST and indicate the order in which you selected the edges

- b- Use Kruskal's algorithm to obtain an MST. Show the final MST and indicate the order in which you selected the edges

- c- Is the MST in this graph unique? If yes, explain why. If no, show edges that can be part of an MST but don't have to be part of every MST.

Additional Space

Additional Space