Comprehensive Examination
Algorithms
Fall 2015

Short Questions

Answer all the short questions (each carry 5 points)

1. Show that for arbitrary real constants a and b, with $b > 0$, we have $(n + a)^b = \Theta(n^b)$.

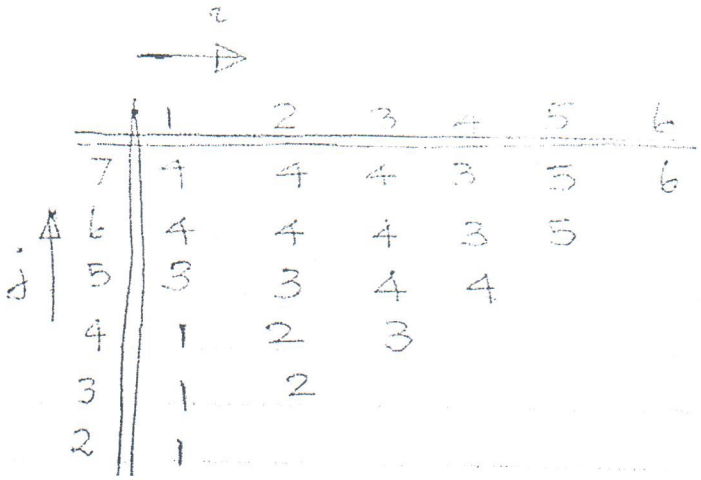
2. Use the recursion-tree technique to derive the tight lower and upper bounds of the recurrence: $T(n) = T(n/3) + T(2n/3) + cn$.

3. What is the time complexity of insertion sort algorithm? Suppose you already know the total number of keys and range of the key values. What would be the best and the worst results you will get when sorting n items. In addition to the previous information, if you already know that the key values are uniformly distributed, what would be your best and worst results. (Make sure you sketch the algorithm and provide rationale of your expected results. Do NOT derive the result)

4. Define minimum spanning tree. What are the salient features of a minimum spanning tree. Name couple of algorithms that will help to find minimum spanning tree from a graph. Out of the two, which one you would prefer and specify why.

5. Mark True or False for each of the following statements:
a. Greedy strategy sometimes find the best or the optimal solution.
b. Dynamic programming will always find the optimal solution even when when principle of optimality condition is not satisfied.
c. Breadth first search is a special case of heuristic search algorithm.
d. Some problems belongs to NP-class can be solved polynomially.
e. Satisfiability problem of propositional calculus is NP-complete.

6. Obtain the optimal parenthesize for a chain multiplication given the S matrix as has been explained in the text book or in the class. (show the working details)



7. Mark True or False against all the following statements:

a. A binary search tree of size N will always find a key at most O(log N) time
b. An optimal binary search is not necessarily a balanced tree
c. A binary heap always maintains a balanced tree as practical as it can be.
d. To implement a priority queue binomial heap is preferred over binary heap.
e. A graph formed by strongly connected component nodes, a strongly connected component graph (SCC), is always a minimum spanning tree.

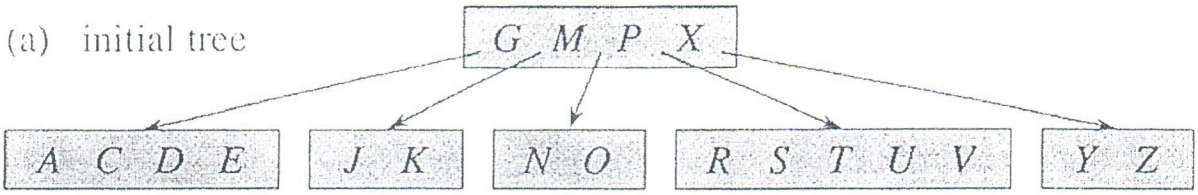
8. For any n-key B-tree of height h and with the minimum node degree of $t \geq 2$, prove that h is no larger than $\log_t \frac{n+1}{2}$. (Hint: consider the number of keys stored in each tree level.)

Long Questions (Answer any 3 and each carries 20 points)

1. The utilization efficiency of a hash table depends heavily on its hashing function(s) employed. Describe with a diagram to illustrate how a multiplication method of hashing works on a machine with the word size of w bits for a hash table with 2^p entries, $p < w$. Explain briefly how Cuckoo hashing works under two hash functions of h_1 and h_2.

2. Given the initial B-tree with the minimum node degree of t = 3 below, show the results (a) after inserting two keys in order: Q then W and (b) followed by deleting two keys in order: Y then T. (Show aggregate result after insertion and another result after deletion.)

(a) initial tree



3. In the following pseudo code,
(a) write the formula to determine number of add operations at line 5 when the algorithm terminates.
(b) What is the space complexity?
(c) Find the following time bounds of this algorithm: upper, lower and tight (must show all the details of your work)

```
Algorithm Count{
1  Cnt=0
2  For i = 1 to n do {
3        For j= i^2 to 1 do {
4           For k= 1 to j do {
5              Cnt = Cnt + 1 }}}
6  }
```

4 A. Describe maximum clique problem.

B. Write a pseudo code to obtain the maximum clique of a graph with N nodes and E edges using generate and test strategy. That is, generate all possible subset of vertices and test whether the subset is a clique. (Make any assumptions explicitly)

C. Find the time and space complexity of your algorithm.

D. Describe 0-1 knapsack problem

E. Show that 0-1 knapsack problem belongs to NP-class

F. Briefly describe a practical way of solving a 0-1 knapsack problem and its time complexity assuming that the knapsack capacity is K and there are M objects. The weight and profit of an object i are denoted by w_i and p_i respectively.