Introduction to Algorithms: 50.004
Singapore University of Technology and Design
18 Dec 2015 3:00-5:00pm

Final Exam

i. This is an open book and open notes exam.
ii. No computer, smartphone and Internet access.
iii. Write your answer on the <u>answer booklet</u>. You may request for additional answer booklet if necessary.

Problem 1: Recurrence and Master Theorem (8 points)

1. [4 points] Consider the recurrence $T(n) = T\left(\frac{n}{8}\right) + T\left(\frac{7n}{8}\right) + 4n$. Write down $T(n)$.

2. [4 points] Consider the recurrence $T(n) = 4T\left(\frac{n}{2}\right) + n^2\sqrt{n}$. Then $T(n)$ is:
   A. $\Theta(n^2 \log n)$ B. $\Theta(n^2\sqrt{n}\log n)$ C. $\Theta(n^2\sqrt{n})$ D. $\Theta(n^3)$

Problem 2: Heap (10 points)

Consider the heap that is represented by the array $A = [5, 13, 2, 25, 7, 17, 20, 8]$. Answer the following questions.

Note: Node 1's value is 5; Node 2's value is 13; Node 3's value is 2…etc.

1. [3 points] Does this heap satisfy the max-heap property? List all the nodes (leave nodes are not included) that satisfy the max-heap property?

2. [4 points] What is the running time of **BUILD-MAX-HEAP** for a given array of length *n*? After we apply **BUILD-MAX-HEAP** for this heap, what is the max-heap? (write down the array)

3. [3 points] If we call **MAX-HEAP-INSERT(A, 21)** after we apply **BUILD-MAX-HEAP** on A, what is the new array?

Problem 3: Sorting (10 points)

Choose the sorting algorithm that would be the best (i.e. most efficient) for each scenario in order to reduce the expected running time. Justify your answer briefly.

1. [3 points] You are running a library catalog. You know that the books in your collections are almost in sorted ascending order by title, with the exception of one book which is in the wrong place. You want the catalog to be completely sorted in ascending order.

   A. Insertion Sort B. Merge Sort C. Radix Sort D. Heap Sort
   E. Counting Sort

2. [4 points] You are working on an embedded device (an ATM) that only has 4KB (4096 bytes) of free memory, and you wish to sort the 2,000,000 transactions withdrawal history by the amount of money withdrawn (discarding the original order of transactions).

   A. Insertion Sort B. Merge Sort C. Radix Sort D. Heap Sort
   E. Counting Sort

3. [3 points] To determine which of your Facebook friends were early adopters, you decide to sort them by their Facebook account ids, which are 64-bit integers. (Recall that you are super popular, so that you have many Facebook friends.)

   A. Insertion Sort B. Merge Sort C. Radix Sort D. Heap Sort
   E. Counting Sort

Problem 4: AVL (8 points)

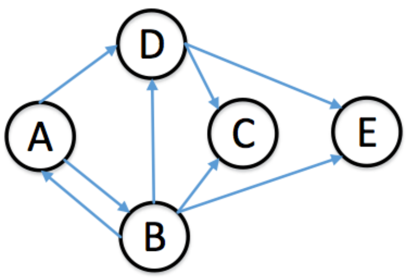
1. [8 points] Show the AVL tree that results after each of the integer keys 9, 27, 50, 15, 2, 21 are inserted, into an initially empty AVL tree. Clearly show the tree that results after each insertion, and any rotations that need be performed.

Problem 5: Hashing (12 points)

1. Use open addressing with linear probing to insert three keys into an empty hash table of size m, assume simple uniform hashing.
   a) [3 points] What is the probability that the second key when inserted required 2 probes?
   b) [3 points] Given that the insertion of the second key required 2 probes, what is the probability that inserting a third key will require 3 probes?

2. [6 points] Let H be a collection of universal functions for a hash table with size $m=n^2$.
   If we select a random $h \in H$ to hash n keys into the table. Justify that expected number of collisions is no more than 1/2.

Problem 6: BFS, DFS (10 points)

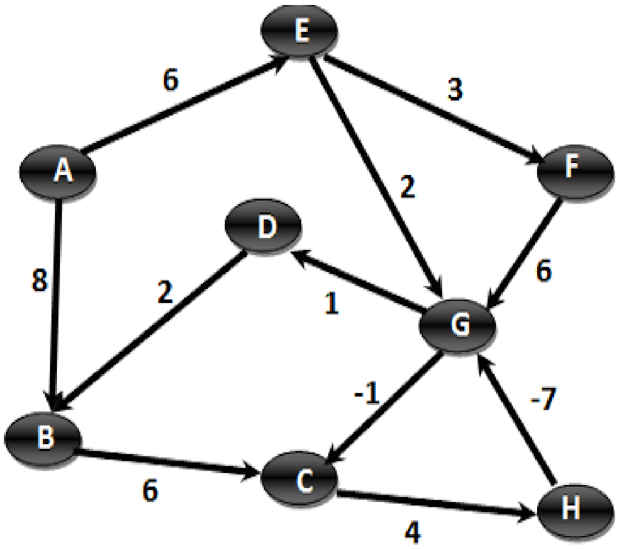


1. Consider the graph above, where a choice exists, use alphabetical order,
   a. [3 points] Give the sequence of vertices visited using a DFS traversal starting at vertex A.
   b. [3 points] Give the sequence of vertices visited using a BFS traversal starting at vertex A.

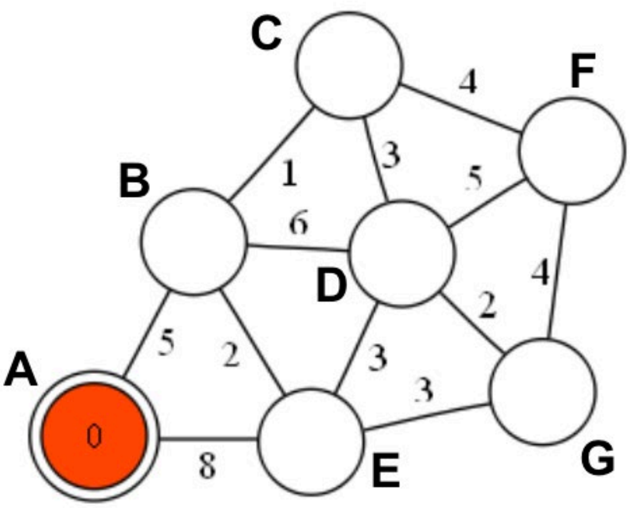
2. Please choose True of False.
   a. [2 points] A directed graph is strongly connected if and only if a DFS started from any vertex will visit every vertex in the graph without needing to be restarted. (True/False)
   b. [2 points] For a directed graph, the absence of back edges with respect to a BFS tree implies that the graph is acyclic. (True/False)

Problem 7: Shortest Path (16 points)

1. [8 points] Find the single source shortest path from A to all the other nodes using Bellman Ford Algorithm.



2. [8 points] Find the shortest path from A to all the other nodes, using Dijkstra's algorithm.



Problem 8: Dynamic Programming (26 points)

1. [2 points] Explain why Dijkstra's algorithm is a Dynamic Programming algorithm.

2. [2 points] Give 1 example, which Greedy Algorithm is more suitable than Dynamic Programming Algorithm. Explain briefly.

3. [2 points] Give 1 example, which Dynamic programming algorithm is more suitable than Greedy Algorithm. Explain briefly.

4. [10 points] Propose a dynamic programming solution for this problem. (Hints: this is a modified knapsack problem)
   - Give n pieces of gold, each have different weight and different value.
   - Each piece of gold only has one copy. No multiple copies.
   - You have two bags of size S1 and size S2.
   - How to maximize the total value of the gold in the two bags?

5. [10 points] Propose a dynamic programming solution for this problem.
   - Given n different values.
   - Divide them into two groups, such that sum(group 1) - sum(group2) is minimized
   - E.g. given {1,3,4,5} the answer is {3,4} {1,5}, the difference of their sum is 1

Bonus Question A: BST (6 points)

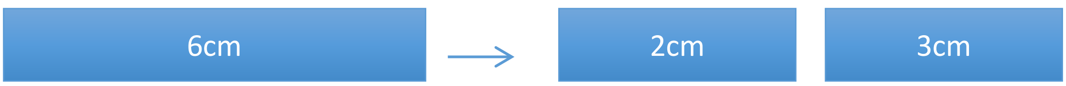
[6 points] Consider a balanced BST with n distinct keys, the rank of a key is its position at which it would be printed in an in-order walk of the tree. Provide an algorithm to retrieve the node storing the key with rank $i, i \in \{1, \cdots, n\}$ in $O(\log n)$) time.

Hint: You may use augmentation: You may keep at node x the value x.size expressing the number of nodes in the subtree rooted at x (including x).

Bonus Question B: Dynamic Programming (5 points)

[5 points] Write down the DP equation of a "modified rod cutting problem", if 1 cm rod will be lost when you cut a rod.

- About rod cutting problem (L11.01):
- Cut a wood into pieces.
- Wood of different length has different value.
- In this question, any cut will result in lost of 1cm. For example:



Bonus Question C: Asymptotic Complexity (9 points)

There are many different algorithms doing matrix multiplication for two $n \times n$ matrices. Answer the following questions.

Input: $A = [a_{ij}], B = [b_{ij}]$.
Output: $C = [C_{ij}] = A \cdot B$.

1. [2 points] **Standard algorithm.** The following is the standard algorithm to do matrix multiplication. What is the running time for this algorithm? Justify your answer.

   for $i \leftarrow 1$ **to** n
       do for $j \leftarrow 1$ **to** n
           do $c_{ij} \leftarrow 0$
               for $k \leftarrow 1$ **to** n
                   do $c_{ij} \leftarrow c_{ij} + a_{ik} \cdot b_{kj}$

2. [5 points] **Divide-and-Conquer algorithm.** Another idea is to use Divide-and-Conquer algorithm. The idea is to divide $n \times n$ matrices into 2×2 matrix of $\frac{n}{2} \times \frac{n}{2}$ submatrices:

   $$\begin{bmatrix} r & s \\ t & u \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \cdot \begin{bmatrix} e & f \\ g & h \end{bmatrix}$$

   and get

   $$\begin{cases} r = ae + bg \\ s = af + bh \\ t = ce + dh \\ u = cf + dg \end{cases}.$$

   Thus, we get a recursive by doing 8 multiplications of $\frac{n}{2} \times \frac{n}{2}$ submatrices and 4 additions of $\frac{n}{2} \times \frac{n}{2}$ submatrices at each step. Write down the recurrence of $T(n)$. What is the running time of this algorithm? Does it improve the standard algorithm?(Hint: addition of two $n \times n$ takes $\Theta(n^2)$.)

3. [2 points] **Strassen's Algorithm,** named after Volker Strassen, implements 2×2 matrices multiplication with only 7 recursive multiplications and gets the recurrence as $T(n) = 7T\left(\frac{n}{2}\right) + \Theta(n^2)$. What is the running time in this case?