No. of Pages: 4

C

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

FIRST SEMESTER M.TECH DEGREE EXAMINATION, DECEMBER 2018

Computer Science and Engineering

Stream(s):

1. **Computer Science and Engineering**
2. ***Information Security***

01CS6105 Advanced Data Structures and Algorithms

Answer *any two full* questions from *each* part
Limit answers to the required points.

Max. Marks: 60 Duration: 3 hours

PART A

1. a. A sequence of Stack operations is performed on a Stack whose size never exceeds k. After every k operations, a copy of the entire Stack is made for backup purposes. Show that the cost of n Stack operations, including copying the Stack, is $O(n)$ by assigning suitable amortized costs to the various Stack operations. 4

   b. A sequence n of operations is performed on a data structure. The *i-th* operation costs i if i is an exact power of 3, and 1 otherwise. Use potential method to determine the amortized cost per operation. 5

2. a. Show the binomial heap that results after each operation listed below: 4
   i. Insert the following numbers, in order, into heap *H1: 12, 7, 25, 15, 28, 33, 41* (show heap *H1* after each step).
   ii. Insert the following numbers, in order, into heap *H2: 18, 3, 37, 6* (show heap *H2* after each step).
   iii. Merge heaps *H1* and **H2**.
   iv. Extract the minimum key from the merged heap.

b. Consider a tree implementation for the union/find problem in which the smaller set is merged to the larger and the name of the set is taken to be the element stored at the root of the tree. Suppose we initialize our sets so that each integer between 1 and 8 (inclusive) is contained within its own set. 5

i. Give a sequence of seven unions that produces a tree whose height is as large as possible. Your answer should be a sequence of procedure calls of the form `Union(a,b)` where a and b are integers between 1 and 8. Draw the resulting tree.

ii. Give a sequence of seven unions, on the original eight sets, that produces a tree of minimum height. Draw the resulting tree.

iii. Explain why both the min- and max-height trees use *seven* unions.

3. a. Describe an algorithm that can *sometimes* detect whether a large integer (say, of 100 or 200 digits) is composite. It is important that your algorithm be more practical than, say, trial division which would run for well over a billion years on a very fast computer with a number of this size. 3

b. Find the greatest common divisor d of 12378 and 3054, and find integers x and y solving the equation $12378x + 3054y = d$. 3

c. Factor 221 using the Pollard's Rho method with polynomial $f(x) = x^2 + 1 mod 221$ and initial guess $x_0 = 2$. 3

PART B

4. a. Let $G = (V, E)$ be a flow network with source s and sink t in which each edge $e \in E$ is restricted to capacity $c(e) = 1$. Further suppose that a maximum flow for G has been computed and an edge is now removed from E. Describe how the maximum flow can be efficiently updated and give the run time of your algorithm. 4

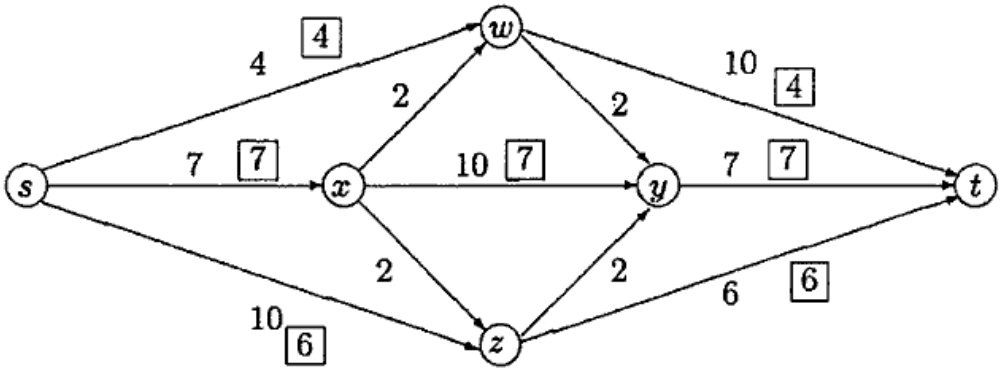
b. Given a string T with only digits (characters '0' - '9'). Suppose we use the Rabin-Karp hash value with mod 113 and 10 as the base. The hash values for the two prefixes of T are $T[1 .. 12] = 100$ and $T[1 .. 6] = 50$. Calculate the Rabin-Karp hash value of $T[7 .. 12]$. ($T[i .. j]$ denotes the substring from index i to index j) 3

c. Define P, NP and NP-complete problems. Give an example for each. 2

5. a. State and prove Maxflow-mincut theorem. 5

b. Show that an algorithm that makes at most a constant number of calls to polynomial-time subroutines runs in polynomial time, but that a polynomial number of calls to polynomial-time subroutines may result in an exponential-time algorithm. 4

6. a. The figure below shows a flow network on which an *s-t* flow is shown. The capacity of each edge appears as a label next to the edge, and the numbers in boxes give the amount of flow sent on each edge. (Edges without boxed numbers have no flow being sent on them.) 5



i. What is the value of this flow?

ii. Is this a maximum *s-t* flow in this graph? If not, find a maximum *s-t* flow.

iii. Find a minimum *s-t* cut. (Specify which vertices belong to the sets of the cut.)

b. Compute the prefix function π for the pattern ***aabaabab*** when the alphabet is $\Sigma = \{a, b\}$. 2

c. Define ***PSPACE*** and ***NPSPACE***. What can you say about the relationship between *PSPACE* and *NPSPACE*? 2

PART C

7. a. A convex polygon in the plane is one in which, for every two points p_1, p_2 on the polygon, the line segment from p_1 to p_2 is contained in (or on the boundary of) the polygon. Given a set P of n points in the plane, no three of which are collinear, the convex hull of P is defined to be the smallest convex polygon containing P. 6

Design an algorithm to compute the convex hull of P that runs in $O(n \log n)$ time. Justify its correctness and running time.

b. Given a graph *G*, a ***CUT*** is a set of edges whose removal splits the graph into at least two connected components. The ***MINIMUM CUT*** is the cut of minimum size. The minimum cut problem is to find a cut of minimum size. Give a randomized algorithm for finding a minimum cut in a given input graph *G*. Find the probability that your algorithm output a minimum cut in the input graph *G*. 6

8. a. i. Explain the difference between a Las Vegas randomized algorithm and a Monte Carlo randomized algorithm. 6

ii. Suppose that a randomized algorithm succeeds (e.g., correctly computes the minimum cut of a graph) with probability p (with $0 < p < 1$). Let ϵ be a small positive number (less than 1). How many independent times do you need to run the algorithm to ensure that, with probability at least $1-\epsilon$, at least one trial succeeds?

b. Consider the following problem "We are given an array of n points in the plane, find whether three points are collinear". Give a $O(n^2*log(n))$ time algorithm that solves the given problem. 6

9. a. Define *RP, co-RP, ZPP* and *BPP* randomized complexity classes. Give at least an example for each. 6

b. Suppose you are given the convex hull of a set of n points in the plane, and one additional point (x, y). The convex hull is represented by an array of vertices in counterclockwise order, starting from the leftmost vertex. Describe an algorithm to test in $O(log\ n)$ time whether or not the additional point (x, y) is inside the convex hull. 6