CS213/293 Data Structure and Algorithms 2023

IITB India

~~Midsem~~

Endsem

Ashutosh Gupta

Compile date: 2023-11-15

Duration : 3 hours Total: 80 marks

- Write all your algorithms in pseudo code
- Do not be unduly worried if you cannot finish the question paper. If you are running short of time, you may describe your algorithm idea in english. However, we deduct marks if any gaps are found.
- Please write detailed formal proofs without significant gaps.

Section A (35 marks)

1. (10 marks) Mark the following statements True / False and also provide justification.
   (a) A graph can have infinite number of vertices.
   (b) A stack does not allow random access to its elements.
   (c) Unordered maps in C++ use red-black trees.
   (d) The minimum spanning tree of a graph is unique if all lengths are unique.
   (e) The number of paths between two vertices in a graph are $O(|V|^2)$.
   (f) A connected graph must have at least $|V|-1$ edges.
   (g) Dijkstra algorithm for shortest path can handle edges with negative length.
   (h) Binary search trees are always balanced.
   (i) KMP is more efficient than trie when there are a large number of searches on a fixed text.
   (j) In C++, if we refer to objects using only `shared_ptr`, there is no possibility of memory leak.
2. (5 marks) Prove that if a graph has an odd cycle then it is not a bipartite graph.
3. (6 marks) In a run of DFS, we classify edges into back, cross, tree, and forward edges. Give conditions in terms of arrival and departure times, and parent relation to identify the class of edges.
4. (7 marks) Let $m = 11$, $h_1(k) = (k \; mod \; 11)$, $h_2 = 6 - (k \mod 6)$. Let us use the following hash function for an open addressing scheme.

$$h(k,i) = h_1(k) + i * h_2(k)$$

What will be the state of the table after each of the following sequence of insertions?
41, 22, 44, 59, 32, 31, 74

5. (7 marks) Let array [A,B,C,D,E,F,G,H,I,J] represents a binary min-heap containing 10 items, where the key of each item is a distinct integer. State which item(s) in the array could have the key with 3rd smallest element.

Section B (45 marks)

6. (8 marks) Given two stacks $S1$ and $S2$ (working in the LIFO method) as black boxes, with the regular methods: "Push", "Pop", and "isEmpty", you need to implement a Queue (specifically : Enqueue and Dequeue working in the FIFO method). Assume there are n Enqueue/ Dequeue operations on your queue. The time complexity of a single method Enqueue or Dequeue may be linear in n, however the total time complexity of the n operations should also be $\Theta(n)$.

7. (8 marks) Let $G = (V, E)$ be a connected labeled graph (no multiple edges between two nodes). Let e be an edge that belongs to a cycle such that its length is larger than all the lengths of edges in the cycle. Prove or disprove: The edge e does not belong to any minimum spanning tree of G.

8. (5+8 marks) The quickselect algorithm finds the kth smallest element in an array of n elements in average-case time O(n). The algorithm is as follows:

Algorithm 3.1: Quickselect(Array G, int k)

```
p := random(1, n)
swap(A, 1, p)
p := partition(A)                                   // Usual quick sort partition
if p = k then return A[p] ;
if p > k then
 | return Quickselect(A[1 . . . p-1], k)
else
 | return Quickselect(A[p+1 . . . n], k-p)
```

(a) Suppose the pivot was always chosen to be the first element of A. Show that the worstcase running time of quickselect is then $O(n^2)$.

(b) Show that the expected running time of (randomized) quickselect is $O(n)$.

9. (6+10 marks) Let us suppose there are n kids in a class and they are planning a field trip. Some are friends and some are not. Their teacher asked each of them for their restrictions. The kids may declare their restrictions in the following four possible ways. (A kid may declare more than one restrictions)

- If X is going, then I will go. (friends)
- If X is going, then I will not go. (enemies)
- If X is not going, then I will go. (vengeful enemies)
- If X is not going, then I will not go. (great friends)

where X is some other kid.

(a) Give an $O(n^2)$ algorithm to check if all restrictions of the kids can be satisfied.

(b) Give an $O(n^2)$ algorithm to find a set of students that may go to the field trip without violating any of the restrictions.

1) bo

4 arrays. 0 1 2 3