Data Structures and Applications (ID2230)

Quiz 2 (Thu, 20 Nov, 2025)

Question:	1	2	3	4	5	6	Total
Points:	10	6	6	5	4	6	37

Duration: **1 hour 45 mins**

Note: BFS = Breadth-First Search, DFS = Depth-First Search, MST = Minimum Spanning Tree, BST = Binary Search Tree, DAG = Directed Acyclic Graph

1. For each of the following, give a one or two sentence reason for your answers. No marks will be given for only the True/False answer. Keep reasons brief but complete.

   (a) [2 points] (True/False) In a Bloom filter storing n elements using k hash functions of m bits using n hash functions, the probability of false positives increases if n is increased.

   (b) [2 points] (Fill in) DFS uses a ____ data structure, and BFS uses a ____ data structure to explore the graph.

   (c) [2 points] (True/False) Depth-first search as described in class will take $O(|V| + |E|)$ time on a directed graph $G = (V, E)$ represented as an adjacency matrix

   (d) [2 points] (True/False) For any connected, undirected, weighted graph G with unique edge weights and $|E| > |V|$, the largest-weight edge e is never a part of any Minimum Spanning Tree (MST).

   (e) [2 points] (Choose and fill in) In the union operation of a Union-Find data structure using union by size, the root of the (smaller/larger) set is made to point to the root of the (smaller/larger) set. This results in a time complexity of _____ for a sequence of k Union operations, starting at n singleton elements.

2. Consider the following undirected graph G: the vertex set is $V = \{A, B, C, D, E, F, G, H\}$. The edges are; A–B, A–C, A–E,B–D, B–E,C–F, C–G, D–H, E–H, F–H, G–H, H–I, H–J. Assume alphabetical ordering when exploring neighbors.

   (a) [2 points] Starting from A, list the BFS discovery order and DFS discovery order of all vertices. Also indicate the level (distance from A) for each vertex in BFS.

   (b) [2 points] Construct and draw the BFS tree and the DFS tree using only the edges selected by the respective algorithms.

   (c) [2 points] Label the non-tree edges of DFS as Back/Forward/Cross edges (in a separate figure to the previous one).

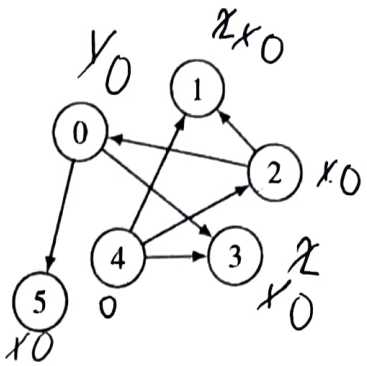
3. Answer the following questions about the unweighted directed graph G below, whose vertex set is $\{0, 1, 2, 3, 4, 5\}$.

   (a) [1 point] Define a topological ordering for a directed acyclic graph.

Question 3 continues...

(b) [3 points] State a topological ordering of G. Then state and justify the number of distinct topological orderings of G possible.

(c) [2 points] State a single directed edge that could be added to G to construct a simple graph (no multiple edges, and no self loops) with no topological ordering. Then state and justify the number of distinct single edges that could be added to G to construct a simple graph with no topological ordering.



4. [5 points] Suppose that you store a set S of n distinct integers in two ways: a min-heap and a balanced Binary Search Tree (e.g. AVL Tree). Write down the time complexities of the implementing the following operations with justification in both the data structures: `Find-Min`, `Insert(k)`, `Delete(x)`, `Search(k)`, `Union(S1, S2)`. Your answers should be in the following format:
`Operation:`_____, `BST-Complexity:` _____, `Heap-Complexity:` _____ Followed by a couple of sentences of explanation for the answers.

   The operations are: `Find-Min`: Outputs the minimum element in S; `Insert(k)`: Inserts the key k into S; `Delete(x)`: Deletes the key in the node/element pointed to by x from S; `Union(S1, S2)`: Given two disjoint sets S_1, S_2, creates the union $S_1 \cup S_2$.

5. Suppose you have two Bloom Filters storing subsets S_1, S_2 of a universe U. Both have been created independently using k random hash functions. Given a key x, we want to know if x is in the symmetric difference $S_1 \Delta S_1$ of S_1 and S_2: i.e. $x \in \{S_1 \setminus S_2\} \cup \{S_2 \setminus S_1\}$. E.g. If $S_1 = \{1,2,3,4\}$, and $S_2 = \{3,4,5,6\}$, then $S_1 \Delta S_2 = \{1,2,5,6\}$.

   We check this using lookups in B_1, B_2 using the following algorithm: Return YES, if the query returns "Definitely YES" for exactly one of the lookups (either in B_1 or B_2).
   Maybe
   Let p_1; p_2 be the probabilities of getting a false positive result for querying in B_1 and B_2 respectively.

   (a) [2 points] Explain why the above procedure may introduce a *false negative* error.

   (b) [2 points] Show that the probability of getting a false negative is at most $\max\{p_1.p_2\}$
   Hint: Consider cases of an element x being contained in S_1, S_2, both or neither, and check when a false negative occurs.

6. A Directed Graph is semi-connected, if for any two vertices u, v, either u is reachable from v or v is reachable from u.

   (a) [2 points] Give an example of a weakly-connected *Directed Acyclic Graph* with a unique source vertex that is not semi-connected (weakly-connected means ignoring edge directions, the graph is connected)

   (b) [4 points] Give an algorithm that runs in time $O(|V| + |E|)$ to check if a DAG is semi-connected. You should argue clearly as to why your algorithm is correct and explain its running time. You may use any algorithm taught in class as a subroutine.

   (c) [4 points (bonus)] Generalize the above to check if a general directed graph G is semi-connected.

End of Questions. All the best!