CS213/293 Data Structure and Algorithms 2024 IITB India
Final

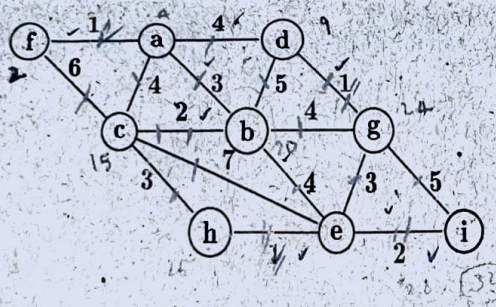
Duration : 3 hours Total: 80 marks

Section A (32 marks)

1. (9 marks) Mark the following statements True / False and also provide justification.
   (a) For any function $f : \mathbb{N} \to \mathbb{N}$, $O(f) \subseteq \Theta(f)$.
   (b) In C++ `std::vector<T>`, the buffer size is doubled whenever the vector is full.
   (c) For `std::map<T,U> m` in C++, `m.at(x)` cannot throw exception for any x.
   (d) C++ uses hash function "SHA1" to implement unordered maps.
   (e) An insertion in red-black tree needs at most two rotations.
   (f) The worst case running time of Heapify is linear.
   (g) The h array in KMP cannot have -10 as an entry.
   (h) The worst case running time of UnionFind with path compression is linear.
   (i) Dijkstra's algorithm does not work for the graphs that have negative edges.

2. (5 marks) What is the probability for the 3rd insertion to have exactly two collisions while using linear probing in the hash table?

3. (6 marks) Run Kruskal's algorithm for minimum spanning tree(MST) on the following algorithm. Show the order of selecting the edges for the MST.



4. (6 marks) Given a directed graph G. Call G^R the graph with all edges reversed (direction). Let $SCC(G)$ be a function that returns the directed graph over the strongly connected components of G, where each strongly connected component is considered a single node. Prove that $SCC(G)^R = SCC(G^R)$.

5. (6 marks) Consider the following modified version of TopologicalSort, where we are using arrival times instead of departure times.

Algorithm 1.1: TopologicalSortArrival(directed graph $G = (V, E)$)

```
1 DFSFull(G, v);
2 if ∃(v, v') ∈ E such that v.arrival ≥ v'.arrival then
3     return "Cycle found: Sorting not possible";
4 return sorted vertices of V in the increasing order of arrival.
```

Prove/Disprove that the above algorithm is correct, i.e., the algorithm returns a valid topological sort of the input directed graph.
In the above, DFSFull is the DFS algorithm from the slides, which runs depth-first search on the entire graph and it sets arrival and departure times of each node.

6. (8 marks) Consider the following tamil word.

காதல்

(a) Give the sequence of character codes to represent the sentence in unicode. Please note that some letters are combination of characters and modifiers.

(b) Let us suppose this word is stored in UTF-8 file format in a file. Given the sequence of bytes stored in the file.

Section B (40 marks)

7. (12 marks) Prove that the average (not worst case) total time complexity for the linked-list based union find algorithm is $O(n \log n)$ for n union operations for the data structure that contains n elements, even if unweighted unions are used.

8. (12 marks) Consider the following sorting primitive:

```
SortTwo(A,i,j) : if( A[i] > A[j] ) swap(A, i, j); }
```

For a given n and A, application of a sequence of the primitive is called sorting network.
For example, the sequence `SortTwo(A,0,1); SortTwo(A,1,2); SortTwo(A,0,1);` is a sorting network, which successfully sorts an array A of size 3. Prove that if a sorting network correctly sorts every sequence of 0's and 1's for a given length, then it correctly sorts every arbitrary sequence of integers for the same length.

9. (8+8 marks) Given an undirected graph $G = (V, E)$. Each edge $(u, v) \in E$ is coloured R or B. A walk in this graph is a sequence of vertices $v_1, v_2, ..., v_n$ such that an edge exists between every v_i and v_{i+1}. Note that vertices (and edges) in a walk can repeat. Given a walk, we can make a string S of the colours of edges taken from v_1 to v_n. Let the string be $C_1C_2...C_{n-1}$ where C_i is the colour of edge (v_i, v_{i+1}).

(a) Given 2 vertices s and t, find if there is a walk from s to t (with v_1 = s and v_n = t) such that S is a palindrome, i.e., the string of colors reads same backwards or forwards.

(b) Find the shortest such walk (n is minimum).

Tamil character codes

Tamil[1][2]

Official Unicode Consortium code chart (PDF)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+0B8x			◌	ஃ		அ	ஆ	இ	ஈ	உ	ஊ				எ	ஏ
U+0B9x	ஐ		ஒ	ஓ	ஔ	க				ங	ச		ஜ		ஞ	ட
U+0BAx				ண	த				ந	ன	ப				ம	ய
U+0BBx	ர	ற	ல	ள	ழ	வ	ஶ	ஷ	ஸ	ஹ					ா	ி
U+0BCx	ீ	ு	ூ				ெ	ே	ை		ொ	ோ	ௌ	்		
U+0BDx	ௐ							ௗ								
U+0BEx							௦	க	உ	ங	ச	ரு	சூ	எ	அ	கூ
U+0BFx	ய	ா	தூ	ஏ	மீ	ஹூ	பூ	ஈ	ஷி	ரீ	நீ					

UTF-8 encoding of character codes

UTF-8 encodes code points in one to four bytes, depending on the value of the code point. In the following table, the characters u to z are replaced by the bits of the code point, from the positions U+uvwxyz:

Code point ↔ UTF-8 conversion

First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
U+0000	U+007F	0yyyzzzz			
U+0080	U+07FF	110xxxyy	10yyzzzz		
U+0800	U+FFFF	1110wwww	10xxxxyy	10yyzzzz	
U+010000	U+10FFFF	11110uvv	10vvwwww	10xxxxyy	10yyzzzz

Original topological sort

Algorithm 1.2: TopologicalSort(directed graph $G = (V, E)$)

```
1 DFSFull(G, v);
2 if ∃(v, v') ∈ E such that v.departure ≤ v'.departure then
3 |  return "Cycle found: Sorting not possible";
4 return sorted vertices of V in the decreasing order of departure.
```