

Ambedkar College of Arts and Science, Wandoor — Class Test 3

Data Structures and Algorithms (MAT5EJ302(1)) — B.Sc. Mathematics Honours, Semester V

Modules I–III (emphasis: Module III)

Name: _____ ID No.: _____ Time: 30 Mins Max Marks: 25

9 questions, 5 marks each — answer any 5 (best 5 count, ceiling 25). Show all working. In every graph question, scan adjacency lists and process ties in alphabetical order.

Q1. Strongly connected components. In a directed graph, vertices u and v are *strongly connected* if there is a path from u to v and a path from v to u (each vertex is strongly connected to itself). A *strongly connected component* is a maximal set of vertices any two of which are strongly connected. Let G have vertices P, Q, R, S, T, U and edges $P \rightarrow Q, Q \rightarrow R, R \rightarrow P, Q \rightarrow S, R \rightarrow S, S \rightarrow T, T \rightarrow S, T \rightarrow U$.

- (a) List the strongly connected components of G . [2 marks]
- (b) List the edges of G that join two different components. [1 mark]
- (c) Can the graph whose vertices are the components (with the edges from (b)) contain a directed cycle? Justify using the definition. [1 mark]
- (d) What is the smallest number of new edges that makes G strongly connected? Give one such set. [1 mark]

[5 M]

Q2. Priority queues in Dijkstra. On a graph with $n = |V|$ vertices and $m = |E|$ edges, Dijkstra's algorithm performs n **deletemin** operations and at most m **decreasekey** operations (all other work is $\mathcal{O}(n)$ and may be ignored). A priority queue may be stored as (i) an *unsorted array* indexed by vertex: **decreasekey** costs $\mathcal{O}(1)$, **deletemin** scans every entry and costs $\mathcal{O}(n)$; (ii) a *binary heap*: both **deletemin** and **decreasekey** cost $\mathcal{O}(\log n)$.

- (a) Total running time of Dijkstra with the array. [1 mark]
- (b) Total running time with the binary heap. [1 mark]
- (c) Which is faster when $m = \Theta(n)$? [1 mark]
- (d) Which is faster when $m = \Theta(n^2)$? [1 mark]
- (e) For what order of growth of m (as a function of n) do the two bounds coincide? [1 mark]

[5 M]

Q3. Breadth-first search. An undirected graph has the adjacency lists

$S: A, B \quad A: B, C, D, S \quad B: A, D, E, S \quad C: A, F \quad D: A, B, F$
 $E: B, F \quad F: C, D, E, T \quad T: F$

- (a) Run BFS from S . Give the distance of every vertex and list the edges of the BFS tree. [3 marks]
- (b) List every non-tree edge together with the two levels it joins. Can a non-tree edge ever join levels that differ by 2 or more? Justify. [1 mark]
- (c) How many distinct shortest S – T paths are there? [1 mark]

[5 M]

Q4. Dijkstra's algorithm. Directed edges with lengths: $S \rightarrow A$ (4), $S \rightarrow B$ (1), $B \rightarrow A$ (2), $B \rightarrow C$ (6), $A \rightarrow C$ (3), $A \rightarrow D$ (7), $C \rightarrow D$ (2), $C \rightarrow T$ (5), $D \rightarrow T$ (1).

- (a) Run Dijkstra from S . Give the vertices in the order they are removed from the queue, with their final distances, and the shortest path to T . [3 marks]
- (b) Construct a directed graph on at most four vertices containing exactly one negative edge (and no negative cycle) on which Dijkstra's algorithm reports a wrong distance, and show where it goes wrong. [2 marks]

[5 M]

Q5. Shortest paths in a DAG. A vertex of a directed graph is a *source* if no edge enters it. A *linearization* of a directed acyclic graph is an ordering of its vertices in which every edge goes from an earlier vertex to a later one. A graph has edges with lengths (lengths may be negative): $S \rightarrow A$ (3), $S \rightarrow B$ (2), $B \rightarrow A$ (-1), $A \rightarrow C$ (-2), $A \rightarrow D$ (4), $B \rightarrow D$ (5), $C \rightarrow D$ (1), $C \rightarrow T$ (6), $D \rightarrow T$ (-3).

- (a) Find a linearization by repeatedly deleting a source (with its outgoing edges); write down the vertex deleted at each step. [2 marks]
 - (b) Find the shortest distance from S to every vertex. [2 marks]
 - (c) Find the *longest* S - T path and its length, using the same graph. [1 mark]
- [5 M]

Q6. Depth-first search on the directed graph with edges $A \rightarrow B$, $A \rightarrow D$, $A \rightarrow F$, $B \rightarrow C$, $B \rightarrow E$, $C \rightarrow A$, $D \rightarrow E$, $D \rightarrow F$, $E \rightarrow F$, $G \rightarrow D$, $G \rightarrow F$.

- (a) Run DFS from A (restart alphabetically) and give the pre and post number of every vertex. [2 marks]
 - (b) Classify every non-tree edge as forward, back or cross. [2 marks]
 - (c) Delete exactly one edge to make the graph acyclic and write down a valid topological order obtained from your DFS numbers (rerun if necessary). [1 mark]
- [5 M]

Q7. Multiplication by halving and doubling. The function `multiply( $x, y$ )` is defined as: if $y = 0$ return 0; otherwise let $z = \text{multiply}(x, \lfloor y/2 \rfloor)$ and return $2z$ if y is even, $x + 2z$ if y is odd.

- (a) Trace `multiply(37, 29)`, giving the argument y and the returned value of every call. [3 marks]
 - (b) For two n -bit inputs, how many recursive calls are made, and what is the total running time in bit operations? [2 marks]
- [5 M]

Q8. Merging sorted lists. The standard *merge* repeatedly compares the front elements of the two lists and moves the smaller one to the output; once one list is empty, the rest of the other is appended with no further comparisons.

- (a) Merge $[2, 5, 9, 14, 20]$ and $[3, 4, 10, 11, 25, 30]$ and count the element comparisons. [2 marks]
 - (b) For sorted lists of sizes m and n , what are the minimum and maximum numbers of comparisons? Give an example list pair for each extreme. [2 marks]
 - (c) What is the worst-case number of comparisons used by merge sort on 8 elements? [1 mark]
- [5 M]

Q9. Growth rates.

- (a) Is $2^{n+1} = \mathcal{O}(2^n)$? Is $2^{2n} = \mathcal{O}(2^n)$? Justify each. [2 marks]
 - (b) Recall that $f(n) = \Omega(g(n))$ means there are constants $c > 0$ and n_0 with $f(n) \geq c g(n)$ for all $n \geq n_0$. Prove that $\log_2(n!) = \Omega(n \log n)$. [2 marks]
 - (c) Compare $n^{\log_2 n}$ and $2^{(\log_2 n)^2}$ for large n . [1 mark]
- [5 M]

— End of Question Paper —