Indian Institute of Technology Kharagpur
Dept. Computer Science and Engineering

Subject No.: CS31005 Subject Name: Algorithms II
Total marks: 60 Duration: 2 hours

Mid Semester Examination

Answer all questions.

---

1. (a) [4 points] Given a network G, a source s and a sink t, as well as a positive threshold integer λ, design an efficient algorithm to find an s-t path where every edge has capacity at least λ.

   (b) [3 points] Continuing from the previous part, give an efficient algorithm to find an s-t path where the minimum capacity appearing on an edge of the path is maximized.

   (c) [3 points] Let G be a network with source s and sink t. Use a max flow algorithm to find a minimum-cardinality set of edges the deletion of which disconnects s from t.

   (d) [5 points] Let G be a directed graph. For a pair of vertices s and t, design an efficient algorithm to find a minimum-cardinality set of vertices the deletion of which will disconnect s from t. [Note: after the deletion of the vertices there could still be paths from t to s]

2. (a) [7 points] Given an empty Fibonacci heap, construct an example of a sequence of insert, decrease-key and extract-min operations such that two cascading cut operations occur. The construction of the heap should be clearly given step-by-step.

   (b) [8 points] For a particular implementation $\mathcal{I}$ of Fibonacci heaps, the code for the Cascading-Cut sub-operation is wrong and none of the steps of Cascading-Cut are executed whenever the Cascading-Cut function is called (all other operations including the Cut operation work as taught in class). Construct an example starting from an empty Fibonacci heap such that using the implementation $\mathcal{I}$ the amortized cost of a Fib-Heap-Decrease-Key operation is no longer $O(\log n)$. The step-by-step construction of the heap

and the argument for why the amortized cost of Fib-Heap-Decrease-Key is no longer $O(\log n)$ should be clearly given.

3. (a) [3 points] Give an example of a connected graph G with at least 10 vertices and a matching M such that there is a flower in G with respect to M.

   (b) [4 points] Show, step by step, how the alternating breadth-first search algorithm in Edmond's blossom algorithm can find the flower of part (i).

   (c) [4 points] What is the worst-case running time of Edmond's blossom algorithm? Prove it.

   (d) [4 points] What is the worst-case running time of Edmond's blossom algorithm when the input graph is a bipartite graph? Prove it.

4. Design a $O(mn)$ time algorithm to compute a blocking flow in the layered graph, defined in the context of Dinic's algorithm for computing a maximum $s - t$ flow.

   (a) [3 points] Clearly explain the algorithm.

   (b) [4 points] Write pseudo-code of your algorithm.

   (c) [4 points] Prove its correctness.

   (d) [4 points] Prove that its time complexity is indeed $O(mn)$.