Problem Set 4

Due: October 12, 1999.

Problem 1. Let's extend a solution from last week. Provide a dynamic graph data structure that supports the following operations in polylogarithmic amortized time:

find-min(r) find the minimum value node in the connected component containing r

decrease-key(v, x) decrease the value of node v to x

delete(e) delete edge e from the graph

insert(e) insert edge e in the graph

Give the best bounds you can on each operation.

Problem 2. Augment a our dynamic connectivity data structure to support path queries: given a pair of points, output a path between them.

(a) Devise a solution that takes $O(n)$ time per path query.

(b) Suppose our graphs have maximum degree d. Devise an *output sensitive* algorithm: if the path length is ℓ, the output time should be $O(\ell d \log n)$. Can you find a faster solution?

Problem 3. Which of the following are true and which are false? Justify your answer with a (short) proof or counterexample.

(a) In any maximum flow, either the flow from v to w or the flow from w to v is 0.

(b) There always exists a maximum flow such that either the flow from v to w or the flow from w to v is 0.

(c) If all arcs in a network have distinct capacities, there is a unique maximum flow

(d) In a directed network, if we replace each directed arc by an undirected arc, the maximum flow value remains unchanged

Problem 4. **Networks with node capacities** In some networks, in addition to arc capacities, each node i other than the source and sink, might have an upper bound, say $w(i)$, on the amount of flow that can pass through it. In these networks we are interested in determining the maximum flow satisfying both the arc and node capacities. Transform this problem to the standard maximum flow problem. From the perspective of worst-case complexity, is the maximum flow problem with node capacities more difficult to solve than the standard maximum flow problem?

Problem 5. Dining Problem Several families go out to dinner together. To increase their social interaction, they would like to sit at tables so that no two members of the same family are at the same table. Show how to formulate finding a seating arrangement that meets this objective as a maximum flow problem. Assume that the dinner contingent has p families and that the ith family has $a(i)$ members. Also assume that q tables are available and that the jth table has a seating capacity of $b(j)$.

Problem 6. A group of students wants to minimize their lecture attendance by sending only one of the group to each of n lectures. Lecture i begins at time a_i and ends at time b_i. It requires r_{ij} time to commute for lecture i to lecture j. Develop a flow-based algorithm for identifying the minimum number of students needed to cover all the lectures.

Problem 7. The census bureau produces a variety of tables from its census data. Suppose that it wishes to produces a $p \times q$ table $D = \{d_{ij}\}$ of nonnegative integers. Let $r(i)$ denote the sum of the matrix elements in row i and $c(j)$ the sum of elements in column j. Assume each is strictly positive. The Bureau often wishes to disclose the row/column sums and certain matrix elements (denoted by set Y) while suppressing remaining elements to ensure confidentiality. However, it may be possible to deduce other elements based on the disclosed ones: d_{ij} can be deduced if only one value is consistent with the disclosed entries.

Develop an algorithm (as fast as possible) for identifying all the deducible elements.

****Problem 8.** Augmenting paths are guaranteed to terminate in finite time with a maximum flow only if the edge weights are rational.

(a) Give a graph on which some augmenting path algorithm fails to terminate it finite time.

(b) (Harder) Give a graph on which some augmenting path algorithm fails to terminate it finite time, and whose limiting flow is not maximum.