CSCI 421 Design and Analysis of Algorithms Spring 2019

Final Exam

1. Recall that inserting into a 2–3 tree temporarily creates 2–3–4 trees. Here is such a 2–3–4 tree.
2. Draw the corresponding left-leaning red-black tree. Mark red links with ‘r’. (Note that it may violate some of the conditions for a valid LLRB.) If you end up drawing several trees, make sure you circle your final answer.
3. Draw the left-leaning red-black tree that results after the insert has been completed.
4. Simulate Dijkstra’s algorithm on the edge-weighted digraph below, starting from vertex 0.

A close up of a map

Description automatically generated

1. Fill in the following table:

|  |  |  |
| --- | --- | --- |
|  | distTo[] | edgeTo[] |
| 0 |  |  |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |

1. What is the maximum number of items in the priority queue?
2. What is the last vertex popped from the priority queue?
3. The square of a digraph G consisting of vertices V and edges E is a digraph G’ such that:

* the vertices in G’ are the same as the vertices in G, and
* two vertices in G 2 are connected by an edge (u, v) if and only if G contains edges (u,w) and (w, v), for some vertex w.

That is, vertices u and v are connected by an edge in G’ whenever G contains a path with exactly two edges from u to v. Describe an algorithm for computing the square of a digraph (represented using adjacency lists). To simplify the problem, you need not remove duplicates from the adjacency lists in G’.

Hint: to compute G’, you need to add an edge to G’ when there is an edge between (u,w) and (w,v), respectively.

1. The FIND-15th problem is to find the 15th smallest item in an (initially unsorted) array. You can implement this easily by sorting the array in O(NlogN) time and returning the item in the 15th position.
2. Given this, which of the following (if any) must be true?

O(NlogN) is a lower bound on FIND-15th problem.

O(NlogN) is an upper bound on FIND-15th problem.

FIND-15th problem must be NP-complete.

1. We can also have an improved algorithm showing O(N) linear time complexity. Describe your linear time algorithm for solving the FIND-15th problem.

Hint: Refer to the lecture slides regarding Quick-select (in Quicksort) or Priority Queues.

1. Consider the weighted undirected graph below.

A close up of a map

Description automatically generated

1. Give the order in which the edges are added to the MST using Kruskal’s algorithm. Start from edge A-B.
2. Give the order in which the edges are added to the MST using Prim’s algorithm (eager version). Start from edge A-B.
3. Consider a chocolate manufacturing company which produces only two types of chocolate – A and B. Both the chocolates require Milk and Choco only.  To manufacture each unit of A and B, following quantities are required:

* Each unit of A requires 1 unit of Milk and 3 units of Choco
* Each unit of B requires 1 unit of Milk and 2 units of Choco

The company kitchen has a total of 5 units of Milk and 12 units of Choco. On each sale, the company makes a profit of

* Rs 6 per unit A sold
* Rs 5 per unit B sold.

Now, the company wishes to maximize its profit. How many units of A and B should it produce respectively?

Model the problem in a linear programming formulation. What are the decision variables? What is the objective function? What are the constraints? What are the restrictions for the decision variables?