Problem Set 3

Both theory and programming questions are due **Thursday, October 6** at **11:59PM**. Please submit your solutions on `http://alg.csail.mit.edu`. The site will prompt you for your answers to the questions, so you do not need to create a solution template. The site will be open for PS2 submissions by Thursday, September 22. You don't need to wait until then to work on your solutions, and we encourage you to start early.

Remember, your goal is to communicate. Full credit will be given only to a correct solution which is described clearly. Convoluted and obtuse descriptions might receive low marks, even when they are correct. Also, aim for concise solutions, as it will save you time spent on write-ups, and also help you conceptualize the key idea of the problem.

We will provide the solutions to the problem set 10 hours after the problem set is due, which you will use to find any errors in the proof that you submitted. You will need to submit a critique of your solutions by **Friday, October 7th, 11:59PM**. Your grade will be based on both your solutions and your critique of the solutions.

Collaborators: Benyamin Jazayeri

Problem 3-1. [45 points] **Range Queries**

(a) [1 point]

Answer: 4

(b) [1 point]

Answer: 3

(c) [1 point]

Answer: 3

(d) [1 point]

Answer: 3

(e) [1 point]

Answer: 6

(f) [1 point]

Answer: 5

(g) [1 point]

Answer: 6

(h) [1 point]

Answer: 5

(i) [1 point]

Answer: 4

(j) [1 point]

Answer: 3

(k) [1 point]

Answer: 2

(l) [1 point]

Answer: 2

(m) [1 point]

Answer: 3

(n) [1 point]

Answer: 4

(o) [1 point]

Answer: 3,4,5

(p) [1 point]

Answer: 3

(q) [1 point]

Answer: 2

(r) [1 point]

Answer: 3

(s) [1 point]

Answer: 8

(t) [1 point]

Answer: 8

(u) [1 point]

Answer: We will first prove an invariant about LCA and then use it to prove it's correctness.

Loop Invariant: node's subtree contains both l and h.

Initialization: node is root so holds.

Maintenance: By the loop condition node.key is either bigger than both l and h or smaller than both l and h. Without loss of generality, if node.key is bigger than both then they are in the left subtree so it takes the smaller value of node.left. So they are still in the subtree rooted by node.

Termination: Case 1: If the loop terminates because node is NIL then l and h do not exist in the tree so the invariant holds. Case 2: If the loop terminates because of the second condition then by the Maintenance clause of the last iteration the invariant holds.

Now we prove that node is the lowest.

Proof by contradiction: Assume for the purpose of contradiction that a node c exists that is lower than node and it's subtree contains both l and h. Assume without loss of generality c is in nodes left subtree. Then $c \leq node$ and so $l, h \leq node$ so node would have taken c.

Problem 3-2. [55 points] **Digital Circuit Layout**

(a) [1 point]

Answer: intersects

(b) [1 point]

Answer: 187590314

(c) [1 point]

Answer: 1,2,5

(d) [1 point]

Answer: 1

(e) [1 point]

Answer: 2

(f) [1 point]

Answer: 4

(g) [1 point]

Answer: 3

(h) [1 point]

Answer: 2

(i) [1 point]

Answer: 2

(j) [1 point]

Answer: count

(k) [1 point]

Answer: 20000

(l) [1 point]

Answer: Implemented using AVL trees.