

CS 362, HW 11

Prof. Jared Saia, University of New Mexico

1. An ordered stack is one that supports the following operations
 - **OrderedPush(x)**: Pops all items off the stack that are less than x , and then pushes x onto the stack
 - **Pop()**: If the stack is non-empty, pops the next item off the stack, otherwise returns ERROR

Suppose you implement an ordered stack using a linked list. Show that you can achieve an amortized cost of $O(1)$ for all operations.

2. **Recursion Cat on a Tree** You are given a tree with all nodes colored either red or black. Call a path *valid* if at any step of the path, the number of red nodes visited so far is greater than or equal to the number of black nodes visited so far. A cat (aka “recursion cat”) starts at the root node of the tree and wants to find a valid path to some leaf node.

For each node v , let $f(v)$ be $-\infty$ if there is no valid path to v . Otherwise, let $f(v)$ be the number of red nodes visited minus the number of black nodes for the path ending at v .

- (a) Give a recurrence relation for f . Hint: you may find it useful to let $p(v)$ be the parent of v , for every node v that is not the root.
 - (b) Briefly describe a dynamic program that uses the recurrence above to return a valid path from root to some leaf, if such a path exists.
3. Exercise 25.2-1 in our textbook: “Run the Floyd Warshall algorithm...”
4. **Recursion Cat on a Graph** Now recursion cat wants to find valid paths on any graph. Define a *red cycle* to be a cycle that has more red than black nodes in it. Assume you are given a graph, G , with no red cycles. For any pair of nodes, you want to determine if there

is a valid path from u to v . Taking inspiration from Floyd-Warshall, you first assign labels 1 to n to all n nodes in the graph. Then you consider paths from nodes u to v that visit intermediate nodes with label at most i . For a given path, let the *black excess* of that path be the maximum over all steps of the path of the number of black nodes minus the number of red nodes at any step. For example, a path of the form R, B, R, B, B, B, R, R has black excess of 2.

Define $f(u, v, i, b) = -\infty$ if there is no path from u to v using intermediate nodes of label at most i , with black excess at most b . Otherwise, define $f(u, v, i, b)$ to be the maximum, over all paths from u to v , with black excess at most b that visit intermediate nodes with label at most i , of the number of red nodes minus the number of black nodes in that path. For example, if the only path from u to v has form R, B, R, B, B, B, R, R , then $f(u, v, n, 2) = 0$.

- (a) Write a recurrence relation for $f(u, v, i, b)$. It may help to assume that $-\infty + x = -\infty$ for any value x . Hint: Let the base case(s) be $f(u, v, 0, b)$ for any values of u, v and any b , $0 \leq b \leq n$. It may help to define for a node v , $color(v)$ to be 1 if the node is red, and -1 if the node is black.
- (b) Briefly describe a dynamic program that uses the recurrence above to determine if a valid path exists from u to v for every u and v . What is the runtime as a function of n , the number of nodes, and m the number of edges?