

# Mark + Sweep Algorithm

Phase 1: Mark all reachable  
cells

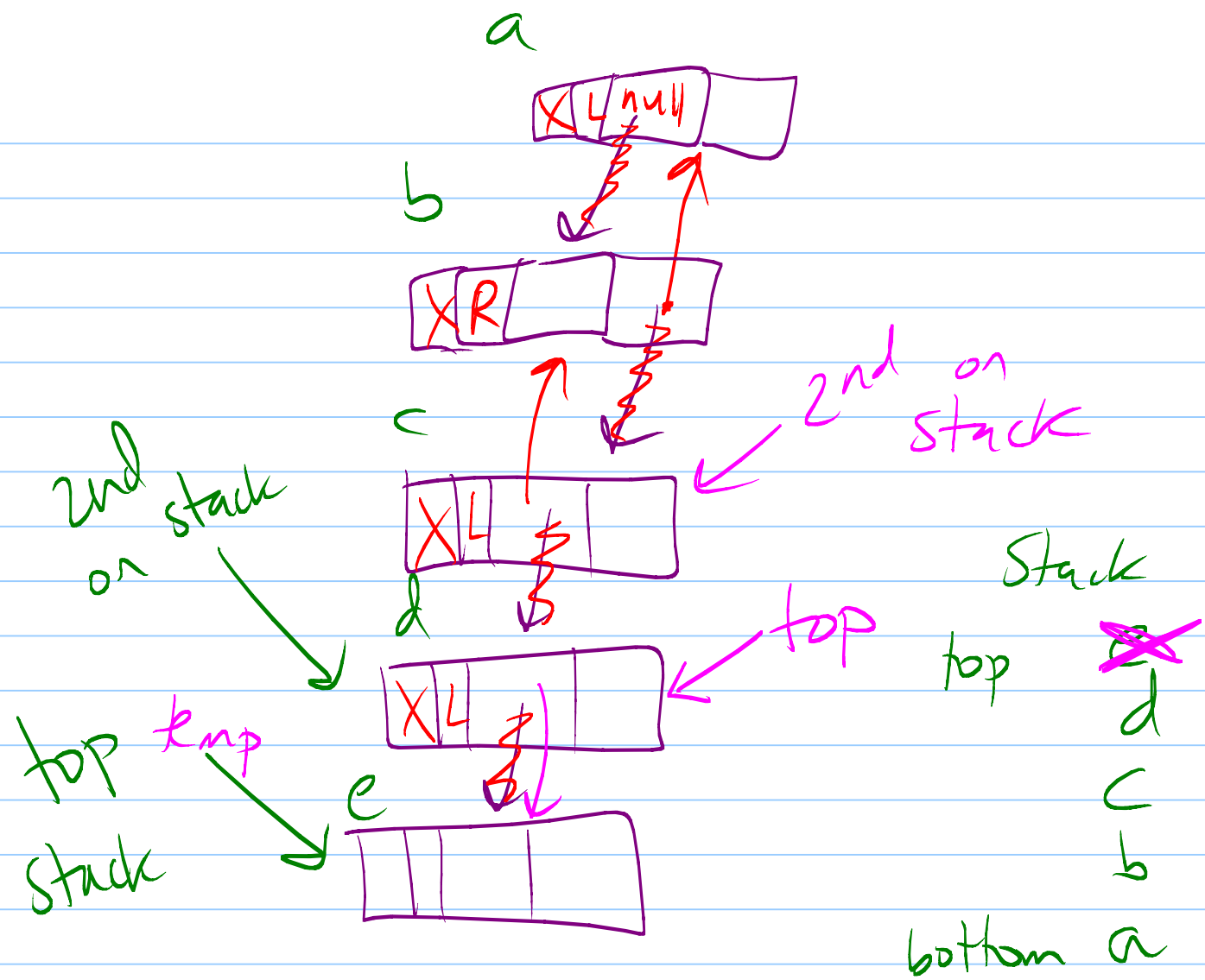
← discovered = true  
means marked

Phase 2 Sweep (go through) all  
of memory + any cell unmarked  
is placed on front of free list

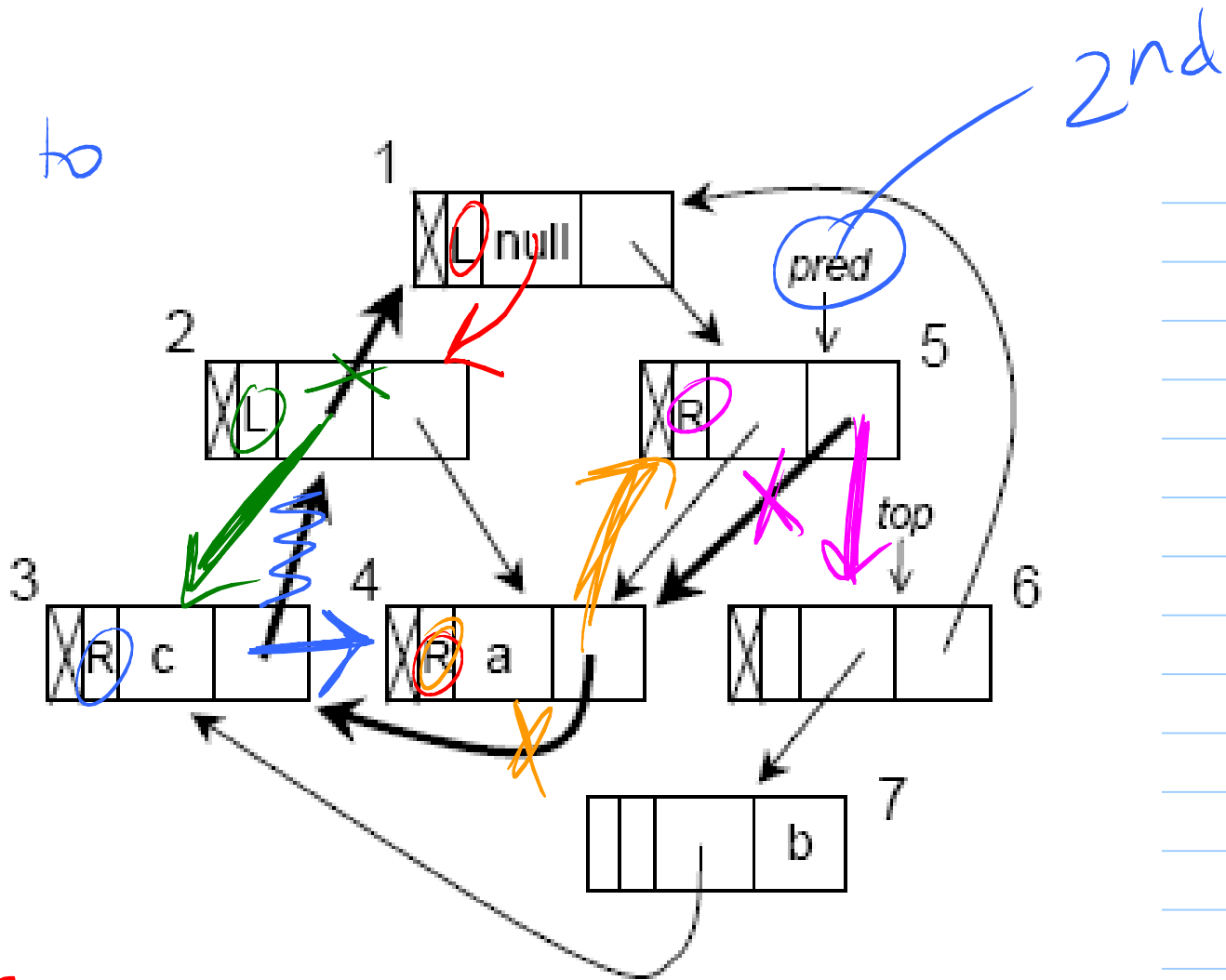
Should we use bfs or dfs  
to minimize space?

Queue (bfs) or stack (dfs)  
need enough space to hold  
a queue/stack as big as  
memory





Restore memory to how it began



Stack 6, 5, 4, 3, 2, 1

## Summary

no external  
stack



Space for in-place DFS  
is 2 bits per memory cell  
(mark bit, back bit) + 3  
L/R

global vars (top, pred, temp)

Time Complexity  $O(M+A)$

$M$  be the # memory cells

$A$  be the # of accessible (reachable) cells

$M-A$  cells of garbage

Mark Phase:  $O(A)$

Sweep Phase:  $O(M)$

each accessible cell  
has at most 2  
edges