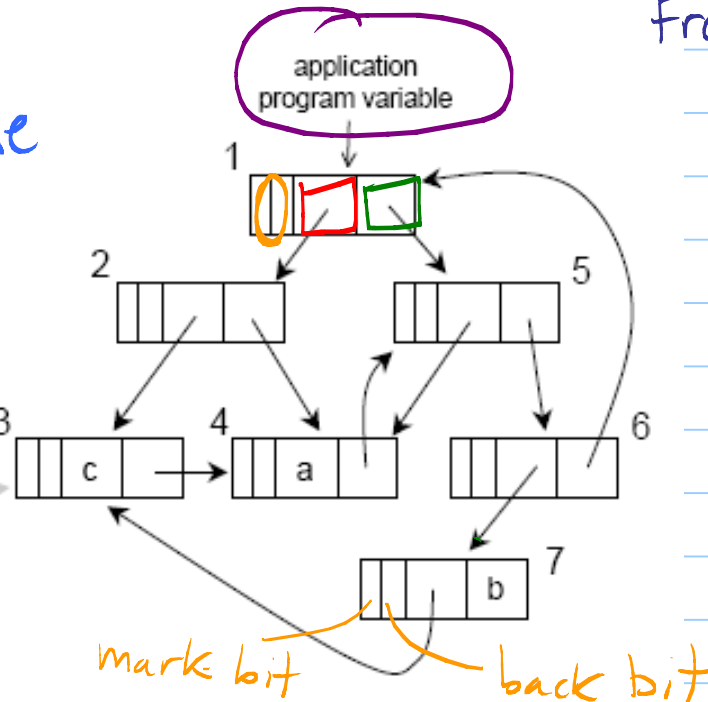
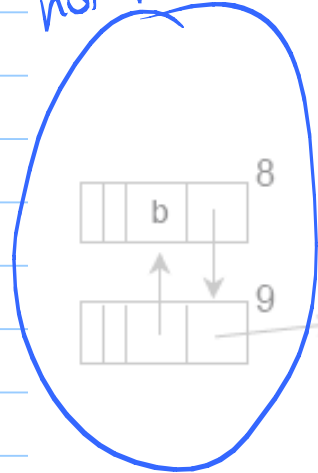


# Garbage Collection

Note Title

12/6/2007

Garbage, not reachable



Before garbage collection

Free list → 10 → 11 → 14 → 13 → 12

Free list

	<u>left</u>	<u>right</u>	
0	10	1	
1	2	5	null
2	3	4	-
3	c	4	
4	a	5	
5	4	6	
6	7	-	
7	3	0	
8	b	9	
9	8	3	
10	11	-	
11	14	-	
12	-	-	
13	12	-	
14	13	-	

How can you efficiently find garbage so you can put it on the free list?

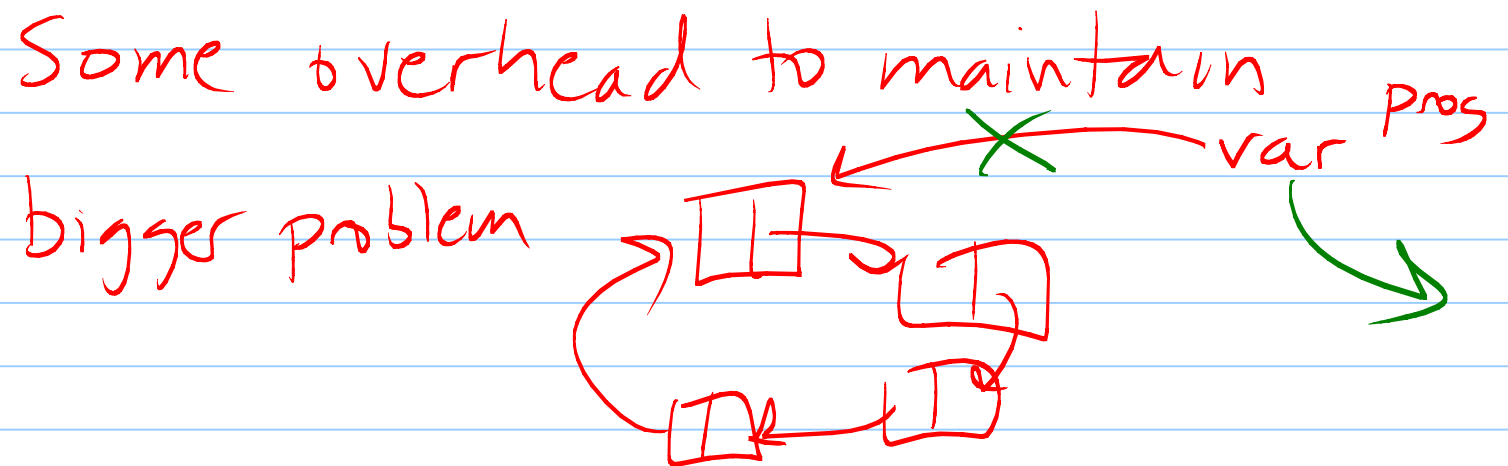
We can find what program can reach & then rest is garbage.

Simple idea: reference counts

Could we just keep track of # refs to a cell?

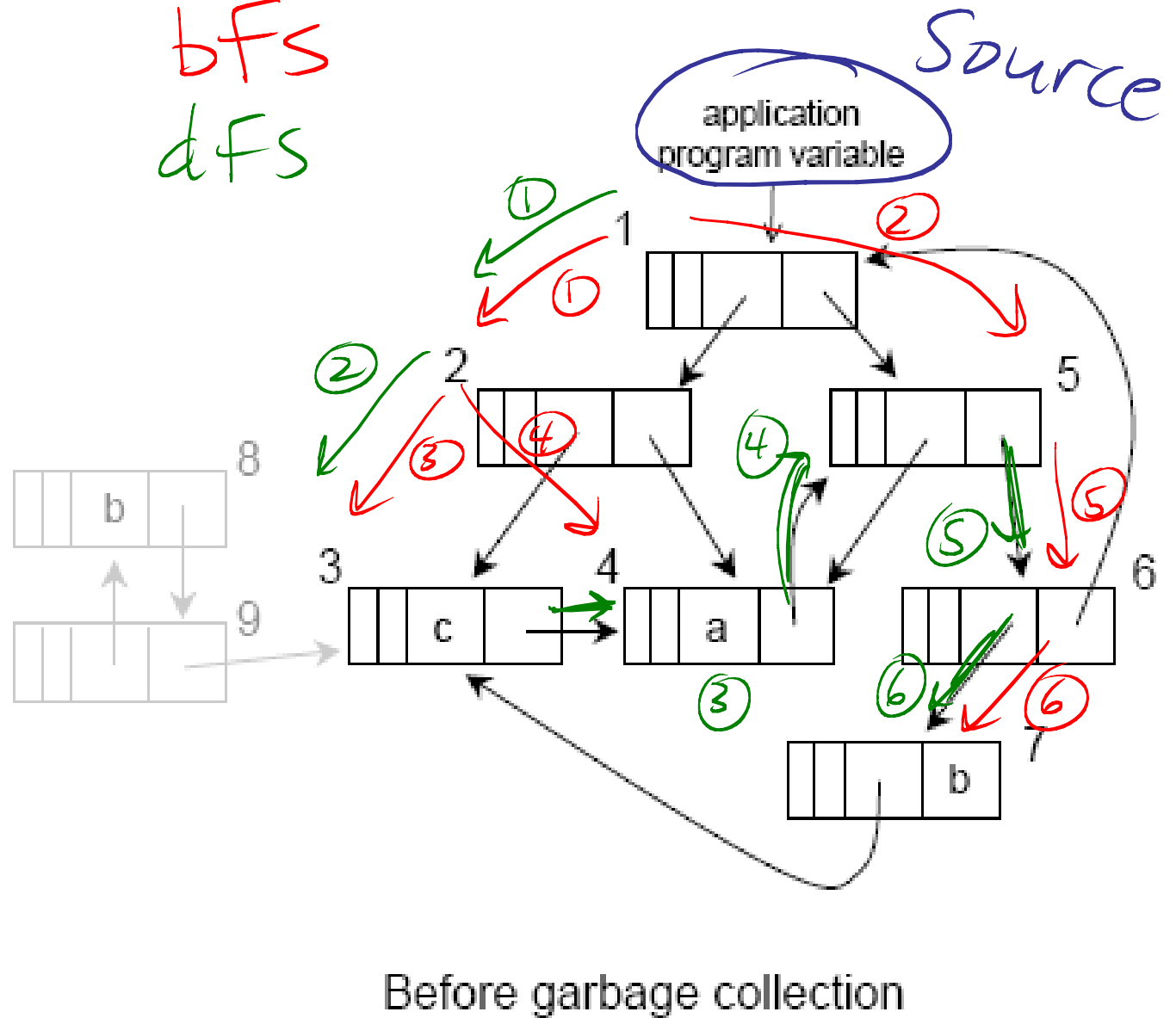
IF ref count is 0 then cell  
is garbage & can be placed on  
free list.

What's the problem?



breadth-first search  
depth-first search can both find all vertices reachable from a given source

bfs  
dfs



Run garbage collection alg when there's nothing on free list.

Important to minimize space used by garbage collection alg.