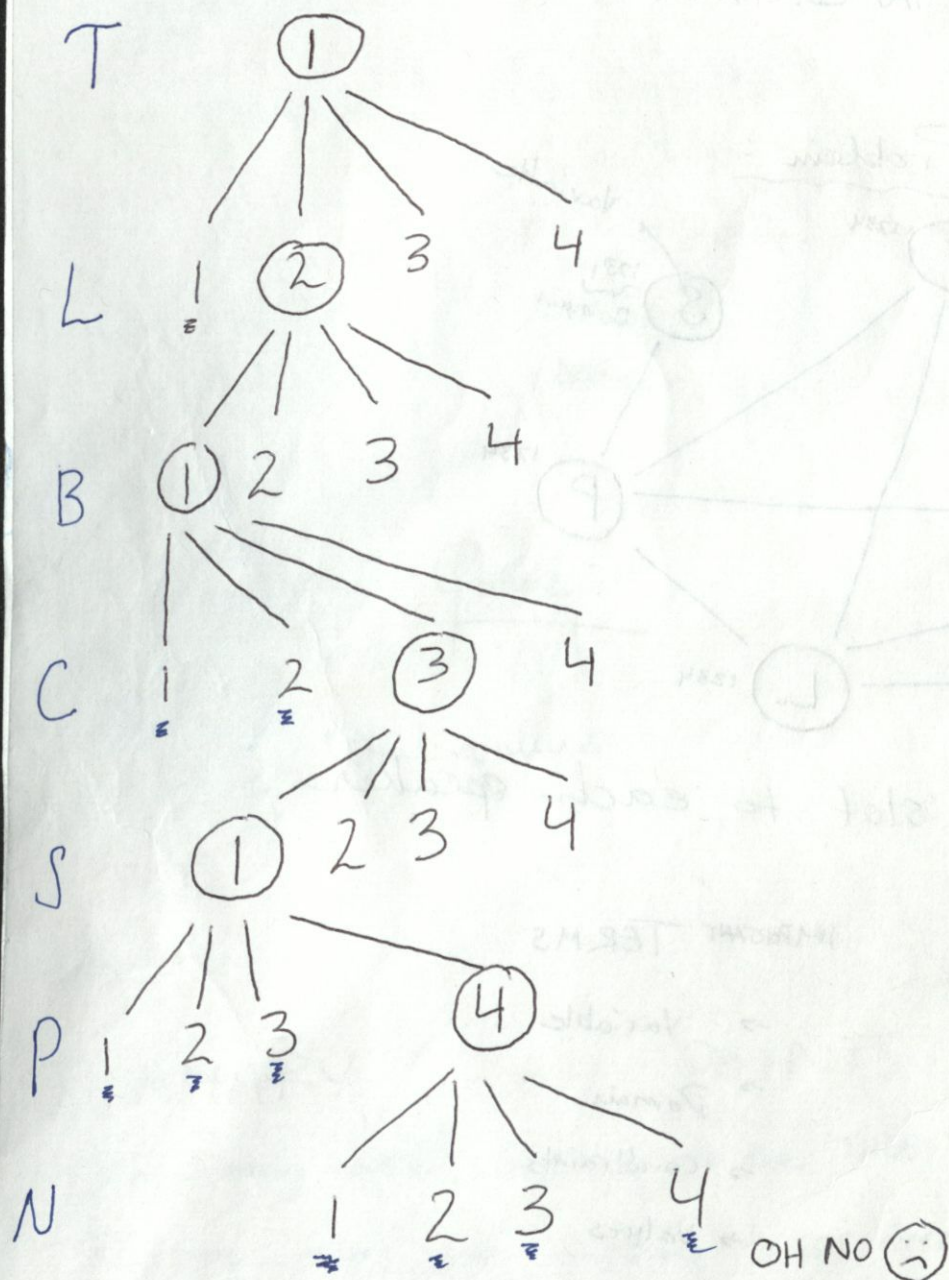


# 1. DFS

- make assignment : (assign a value to a variable)
- check for contradictions with past assignments

Tree: Domain



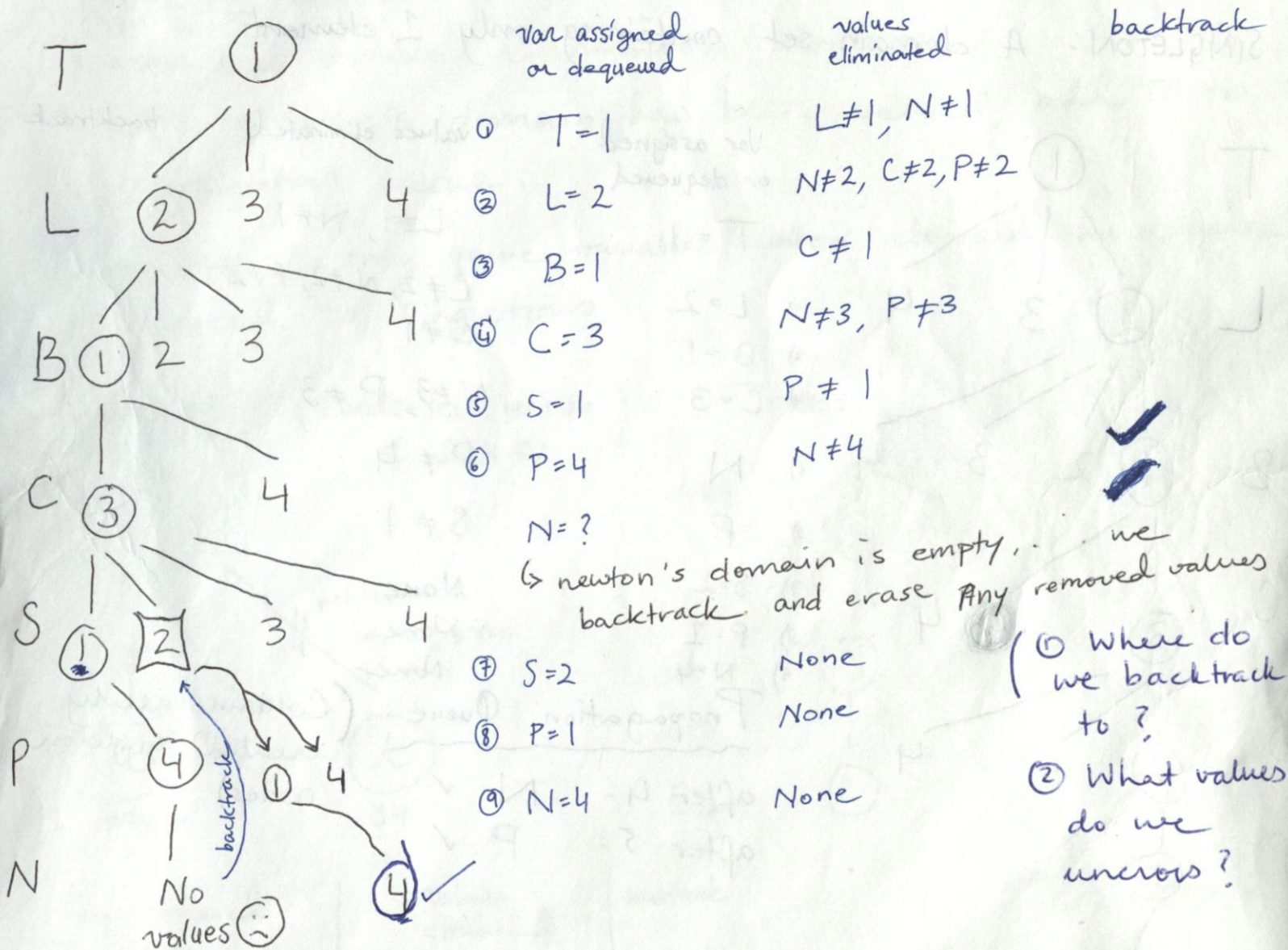
- Clearly, this set of values does not work. ∴, here we would backtrack up to S and instead choose S=2 and then check if that solves the problem.



# S + FC

check all neighbors of assigned variables and cross out inconsistent values from neighbors' domains (only neighbors that have not been assigned)  
 → When backtracking, uncross out values  
 \*(uneliminate eliminated values)

NOTE: When a domain is reduced to nothing, BACKTRACK.



Final Assignment: T=1, L=2, B=1, C=3, S=2, P=1, N=4

\* Now, we don't have to bother checking for contradictions with past assignments.

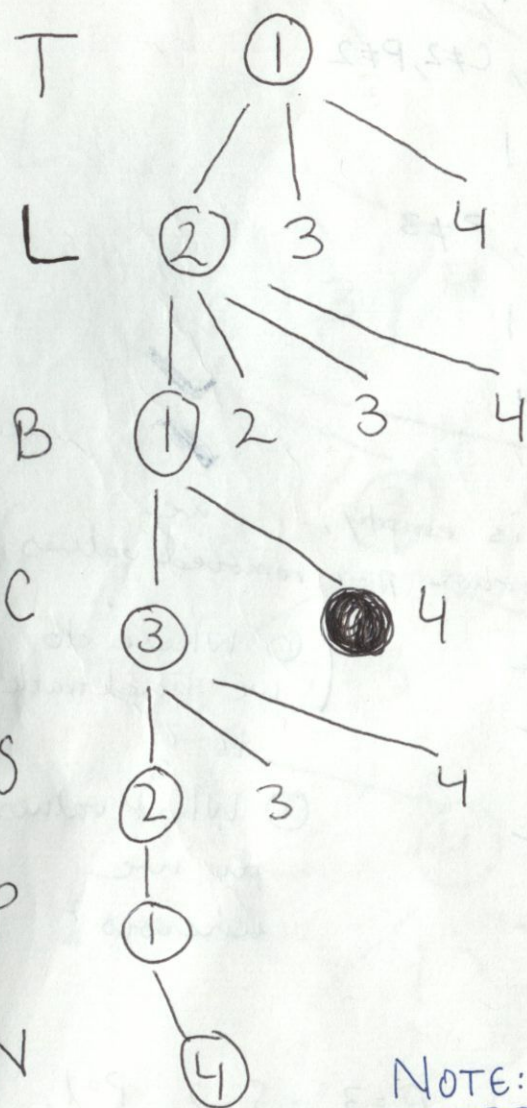


~~Handwritten scribbles and a star symbol.~~

① tree, table, & propagation

- check for NEW singleton domains
- if one is found, DON'T assign yet, but use it to eliminate from neighbors' domains.
- if this creates a new singleton domain, PROPAGATE

SINGLETON: A domain set containing only 1 element



Var assigned  
or dequeued

values eliminated

backtrack

$$1) T = 1$$

2)  $L=2$

3)  $\beta = 1$

4)  $C = 3$

5) N

g) P

7)  $S=2$

8)  $P=2$

9)  $N=4$

$$L \neq 1, N \neq 1$$
$$C \neq 2, N \neq 2, P \neq 2$$
 $C \neq 1$ 
$$N \neq 3, P \neq 3$$
$$P \neq 4$$
 $S \neq 1$ 

None

None

None

Propagation Queue: (Contains newly created singleton nodes)

after 4: N ✓

after 5: P ✓

NOTE:

The reason we only propagate NEWLY created singleton domains is to avoid repeating work. Otherwise, we'd keep propagating a single variable after every iteration.

\* Also, when backtracking, queue is completely cleared  
→ We create a queue after assigning a variable.



## 4. DFS + FC + Prop-any

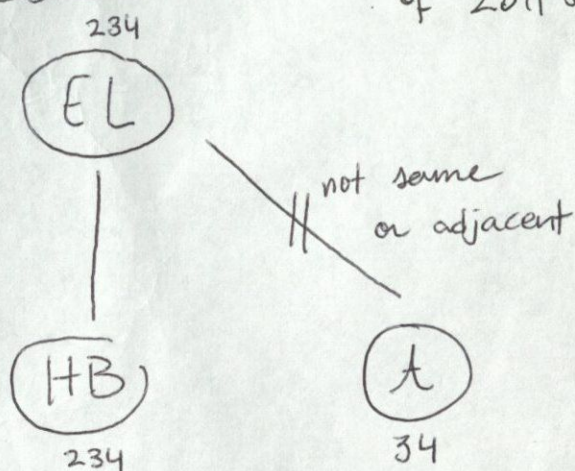
- only interesting when the constraint types are more complicated such as can't be adj.,  $\geq$  etc

Propagating a Variable: (applies to prop-1, prop-any, domain reduction)

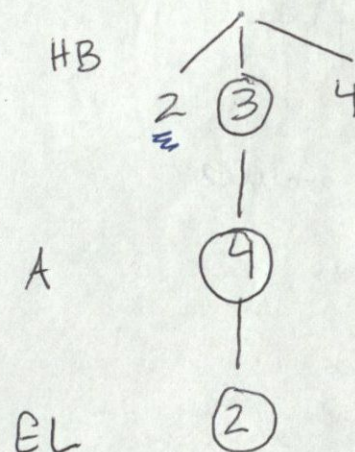
- look @ variable's remaining options
- cross out values in neighbors' domains
- if a neighbor's domain has been reduced, add it to propagation queue.

\* → for prop-1, only queue variables whose domains have been reduced to singletons.

PROBLEM: (Condensed version)  
of 2011 Q2



Tree



| var assigned<br>or dequeued | values<br>eliminated | backtrack |
|-----------------------------|----------------------|-----------|
| ① HB=2                      | EL ≠ 2               |           |
| ② EL                        | A ≠ 3, 4 → ✓         |           |
| ③ HB=3                      | EL ≠ 3               |           |
| ④ EL                        | A ≠ 3                |           |
| ⑤ A                         | EL ≠ 4               |           |
| ⑥ EL                        | None                 |           |

Prop. Queue:

- after 1, EL ✓
- after 3, EL ✓
- after 4, A ✓
- after 5, EL ✓