# CS2100

## Treaps

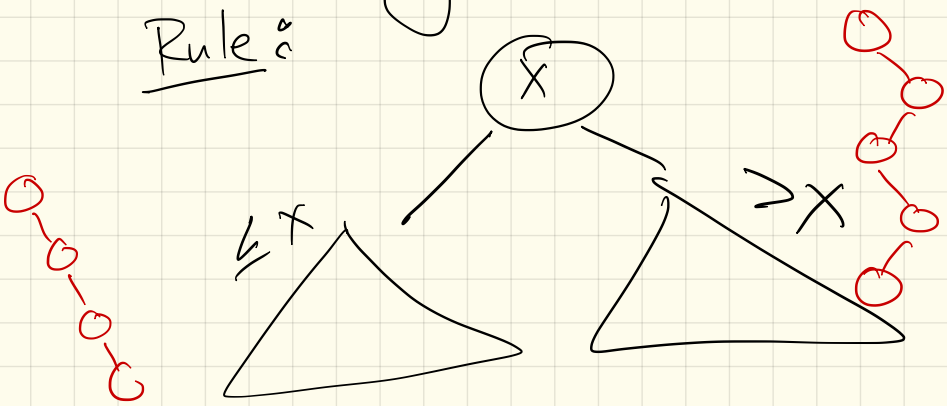
# Recap

- HW: next Thursday
- Friday is review, following Monday is test
- Midterm content: through AVL trees

# Last time

End of Binary Search trees

Rule:




Runtime of find (& insert/delete as a result):

$O(\text{height}(T)) = O(n)$

AVL trees:

H-B property

At every node x, heights of x's children must be within $\pm 1$.

Result: $\text{height} \leq 2\lceil \log_2 n \rceil$

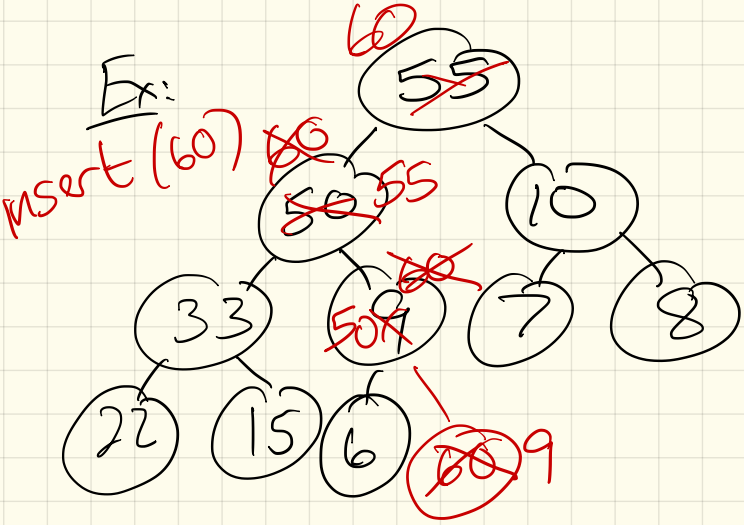
find (& others): $O(\log n)$

# Another recap:

**Heaps:** (not BSTs) priorities

At every node, the key value will be ≤ key at either child

Ex:




**Runtimes:** $O(\log n)$

# Treaps: a new binary tree structure

**Goal:** Each node will contain a *value* (like a BST) and a *priority* (like a heap).

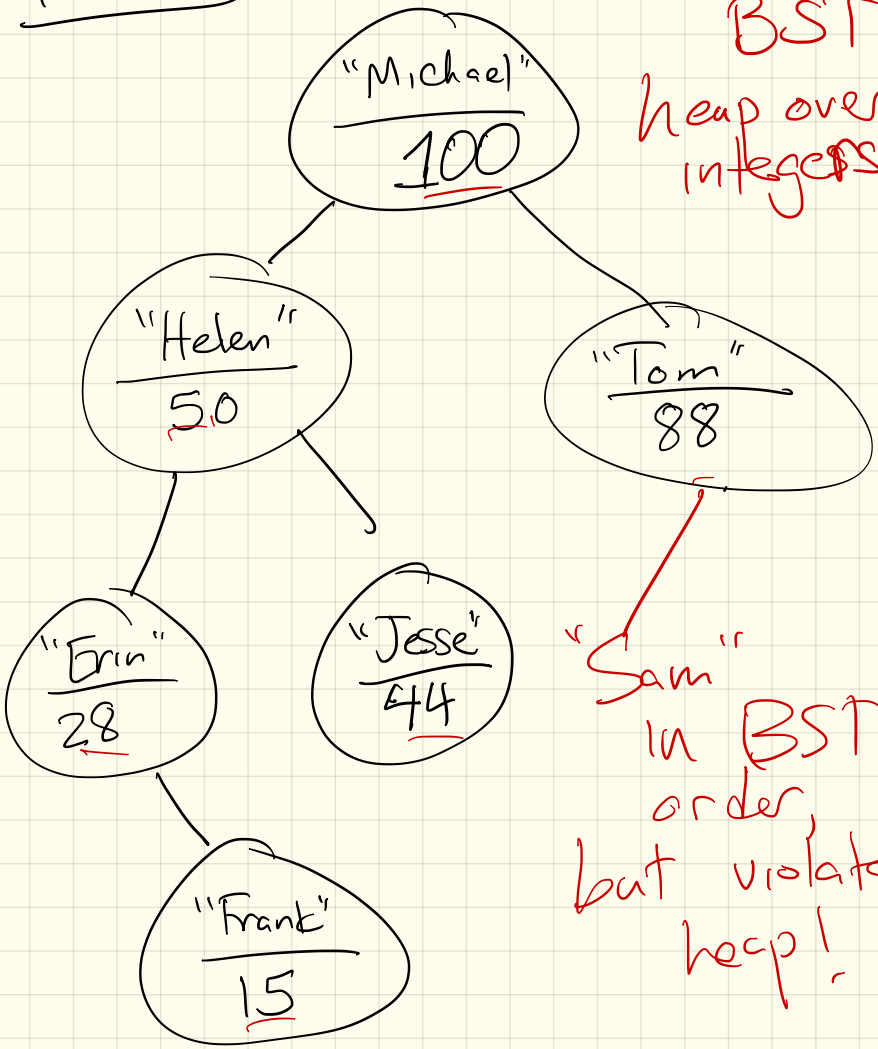
- BST over values
- heap over priorities

**Ex:** Suppose values are names and priorities are integers.

Both can be "sorted":

- values/names have alphabetical order
- integers (obviously)

Picture:



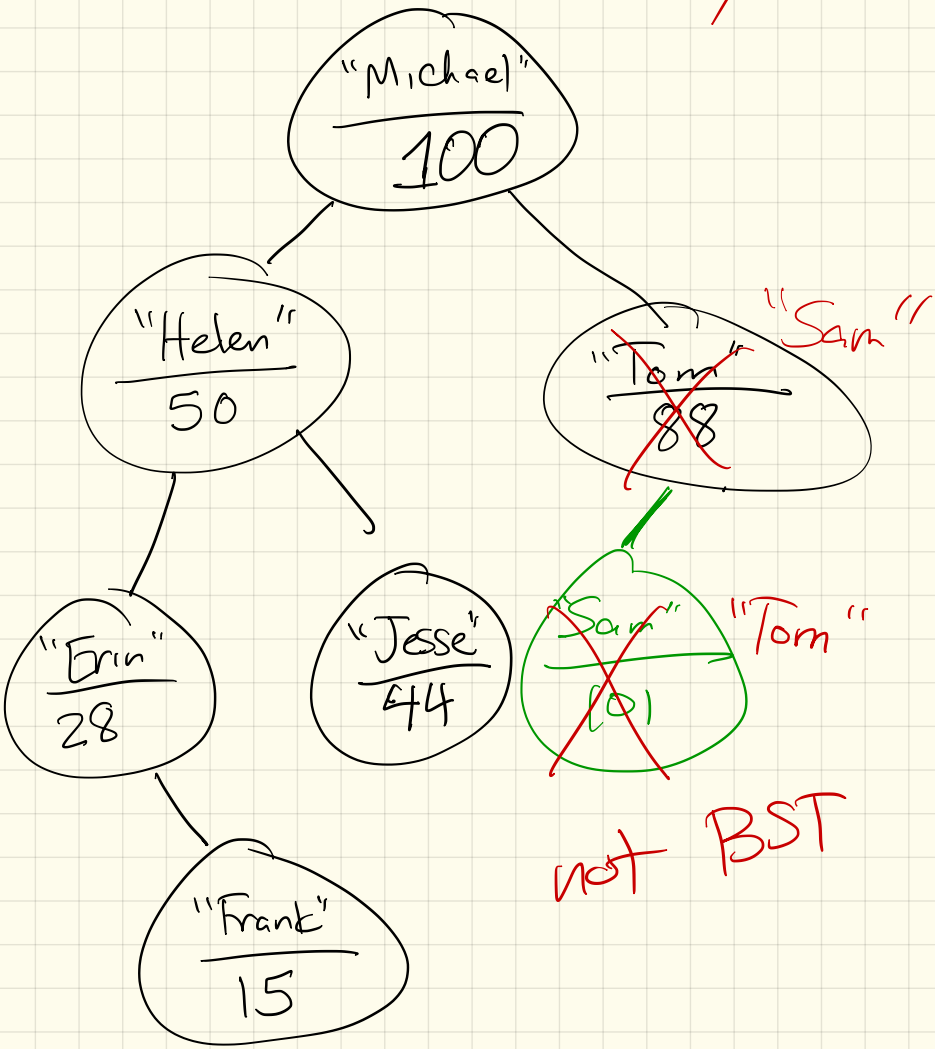

names: BST ✓

heap over integers

in BST order, but violates heap!

how: insert("Sam", 101)

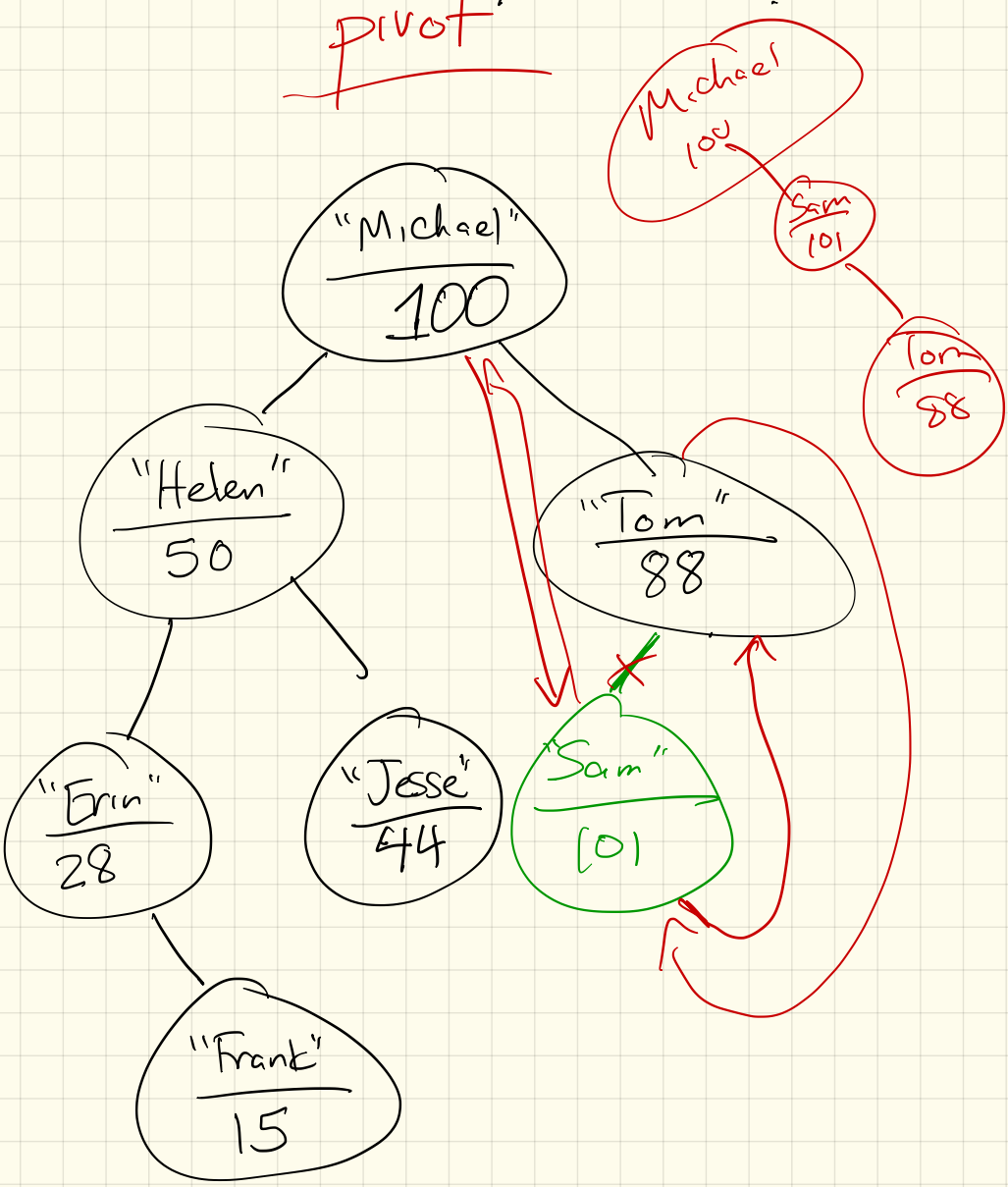
In heaps, we "bubbled" up.
Can we do that here?

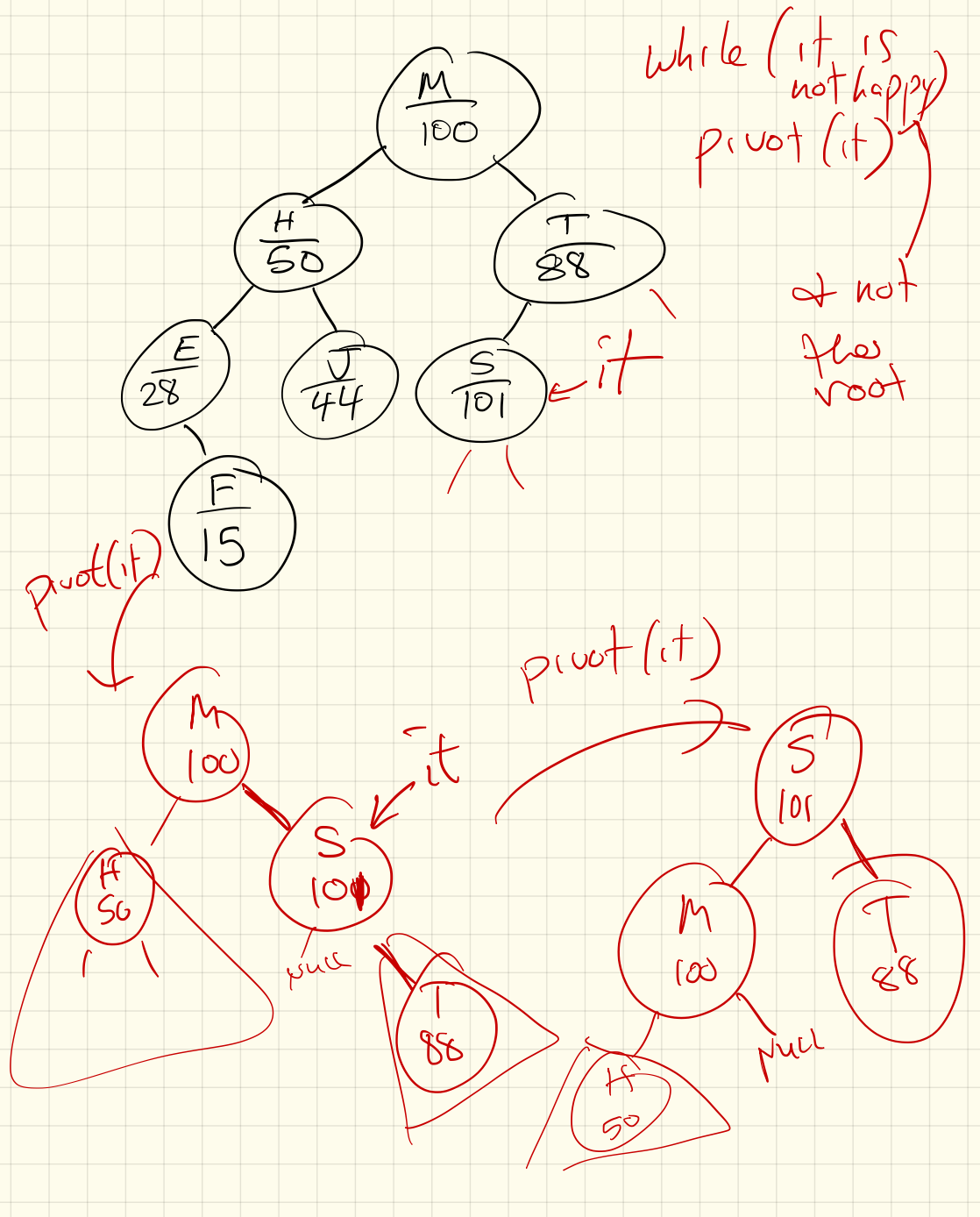



not BST

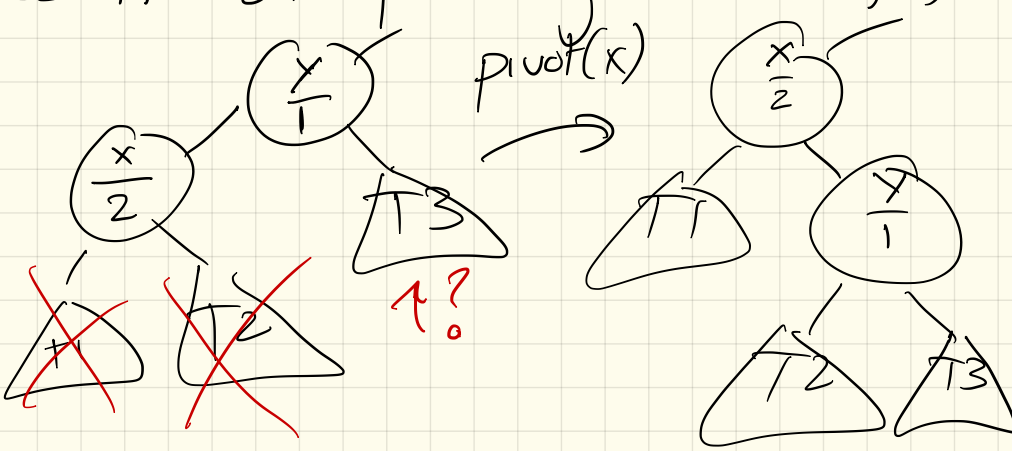
Well - can't violate BST!

What did we do to move things around in AVL trees?




while (it is not happy)
pivot(it)
& not the root
M
100
H
50
T
88
E
28
J
44
S
101
it
F
15
pivot(it)
M
100
H
50
S
100
it
null
T
88
pivot(it)
S
101
M
100
T
88
H
50
null

Result of pivoting: insert(x, 2)




Clearly, we're still a BST!

Can T1, T2, or T3 be non-heaps after this?

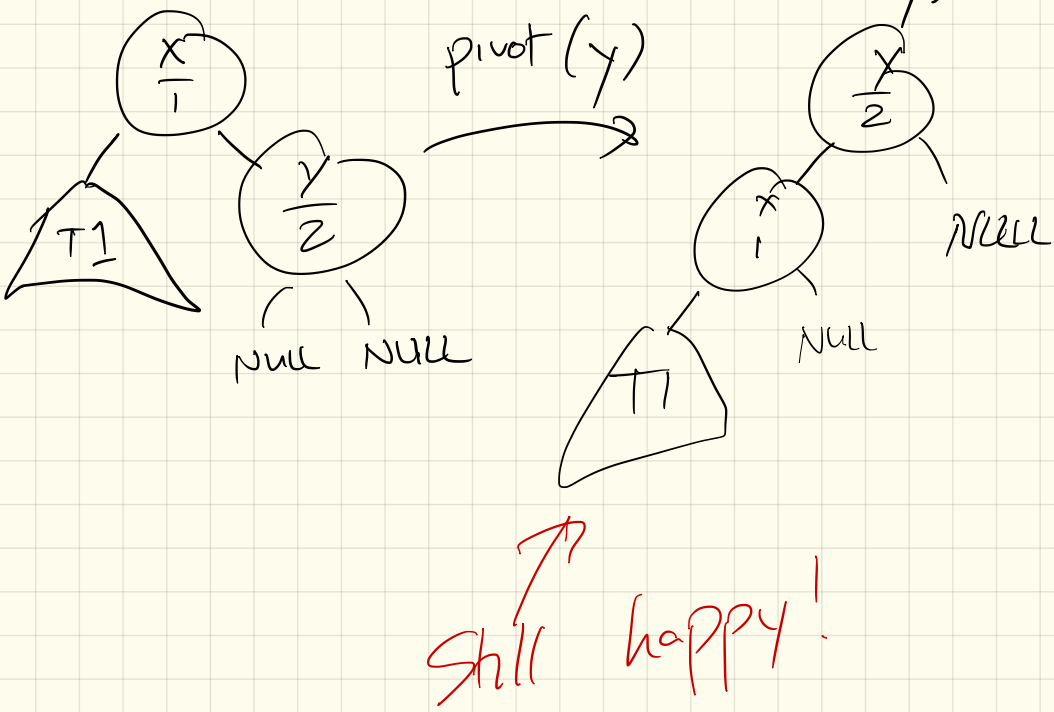
(Note: just inserted x, so what are T1 + T2?)

↳ NULL

T3?: T3 had priorities all ≤ y's priority

→ y is still above, so still in heap order

Same for other case: insert(x, 2)




Still happy!

Result:

data

aux

Insert (val, key):

Run BST insert (on data)

Save its location, it

while ((it.priority > it's parent's priority)

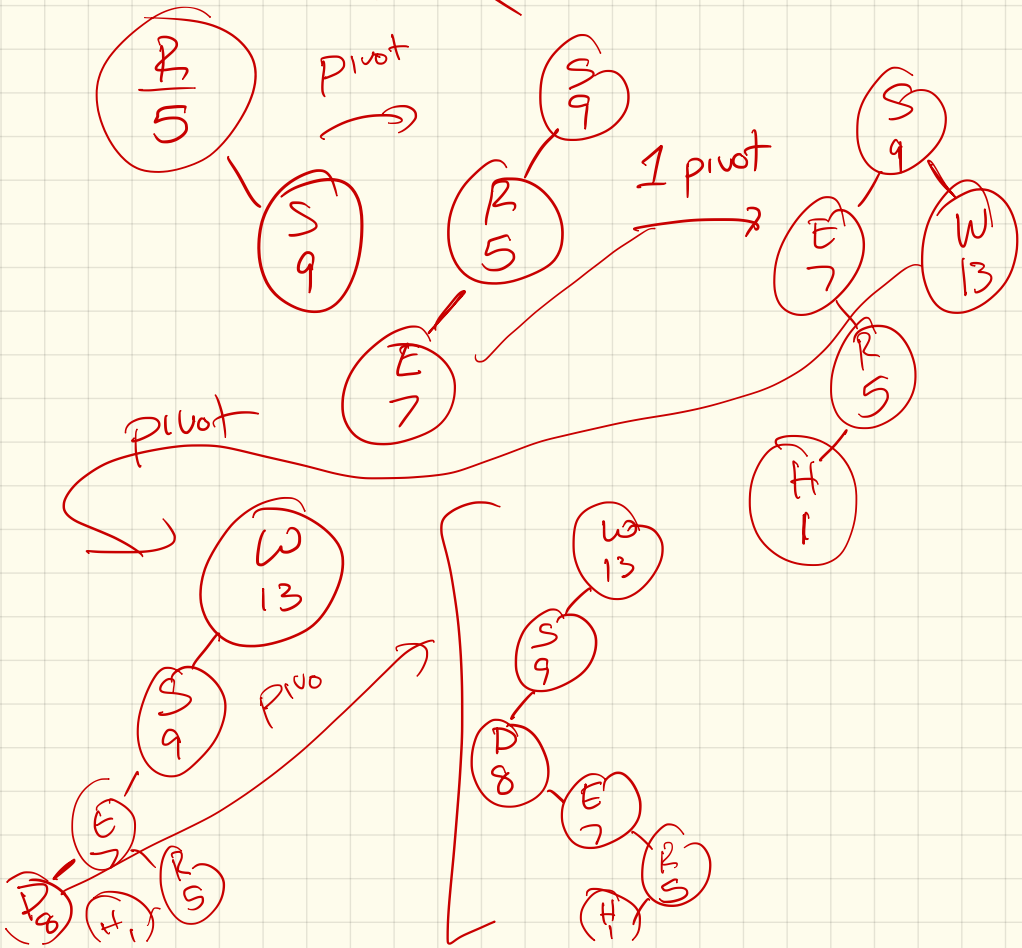
and

it != root())

pivot(it)

Example: Insert:

(R,5), (S,9), (E,7), (H,1),
(W,13), (D,8), (J,2), (K,4), (P,11)



Observation:

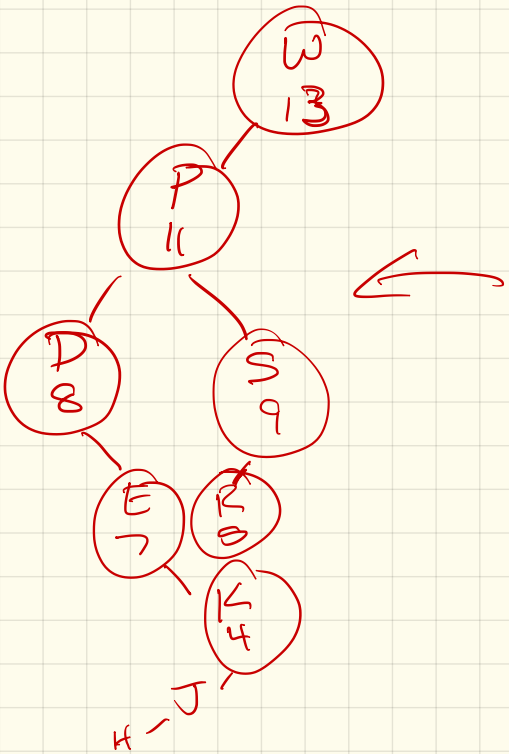
Take a step back.

What must the root be?

Example: Insert:

(R,5), (S,9), (E,7), (H,1),

(W,13), (D,8), (J,2), (K,4), (P,11)



## Observation

Treaps are unique

(BSTs & heaps are not)

(& AVLs)

→ This is like giving 2 traversals.

Next: • remove

• run times

(randomized)