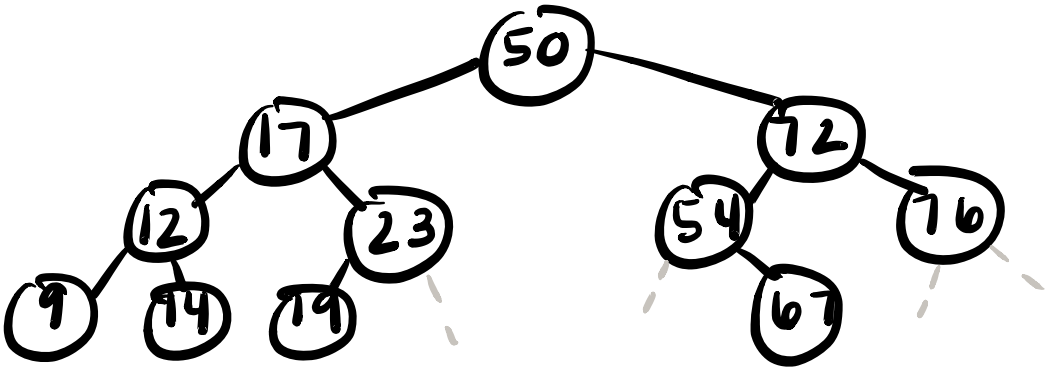
SUCCESOR = smallest value greater than x
aka the next # after x

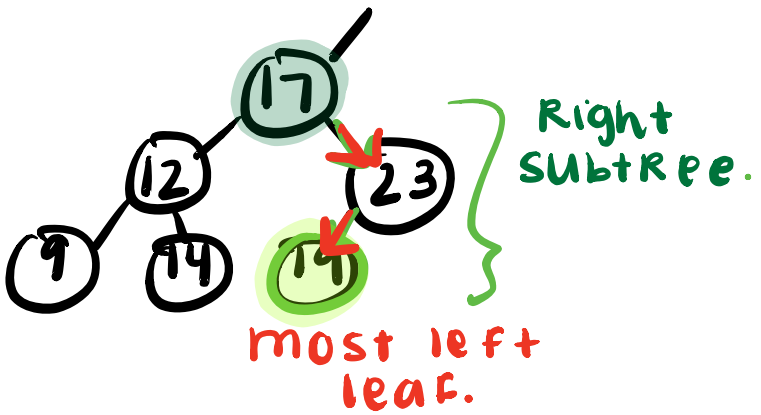
PRedecessor = greatest value smaller than x
aka the # before x



SUCCESSOR

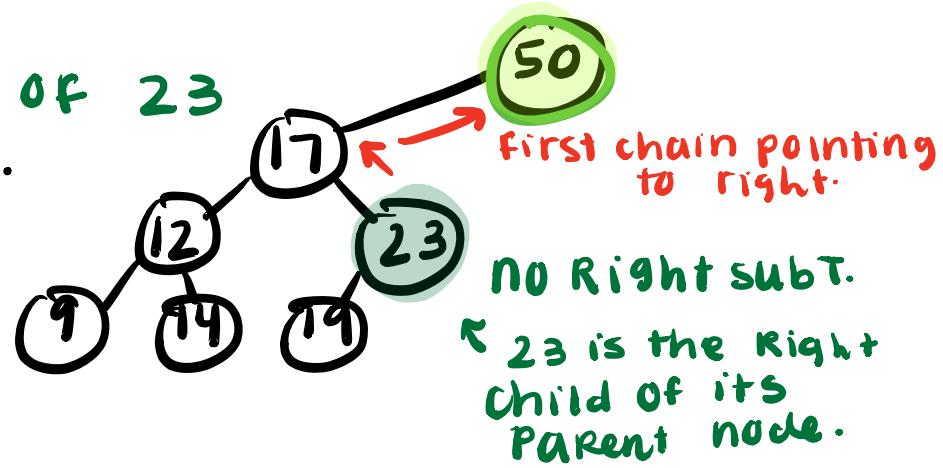
- IF node x has a Right subtree, then the successor is the left most leaf. OF that Right subtree

Ex) successor of 17
= 19.

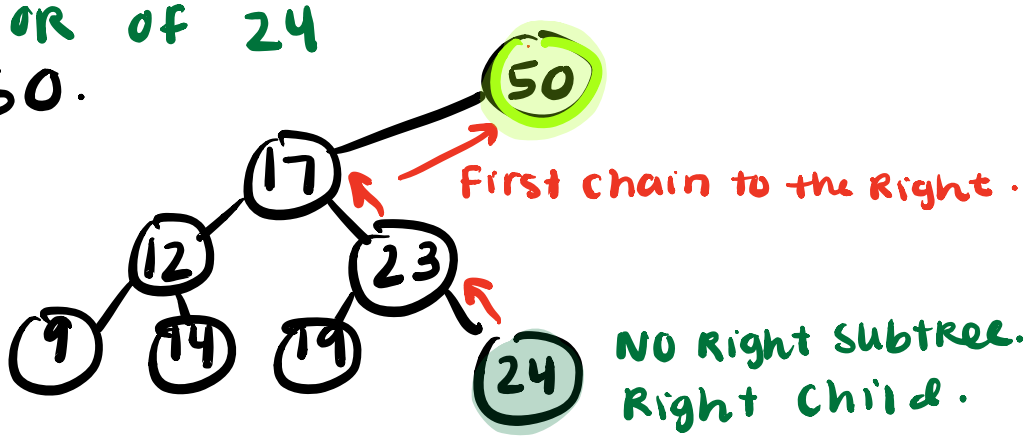


- If there is no Right subtree, and the node is a Right child, then you have to go up the tree and look for the chain that goes to the Right

Ex) Successor of 23
= 50.



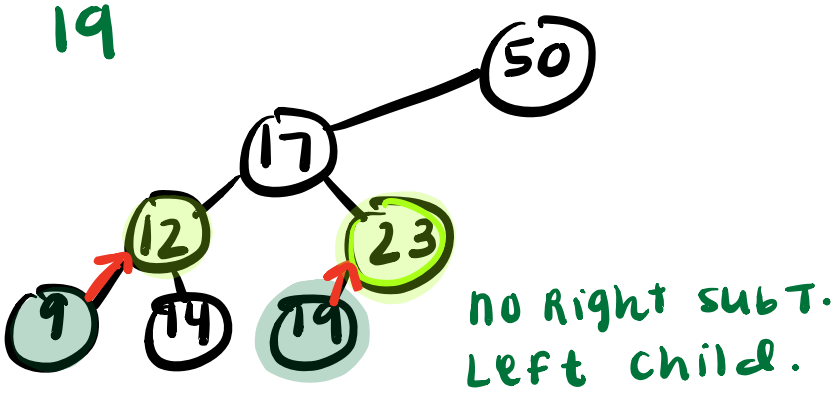
Ex) Successor of 24
= 50.



- If there is no Right subtree, and the node is a left child, then the successor will be the parent of the node.

Ex) Successor of 19

= 23.



Ex) Successor of 9

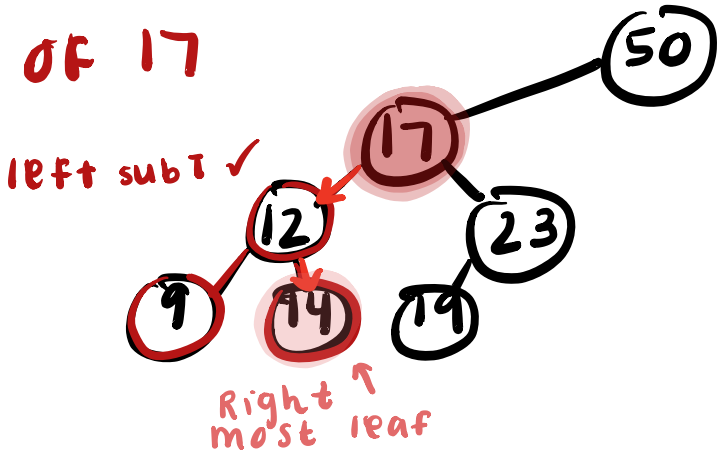
= 9

Predecessor

- If node x has a left subtree, then the predecessor is the Right most leaf. of that Right subtree

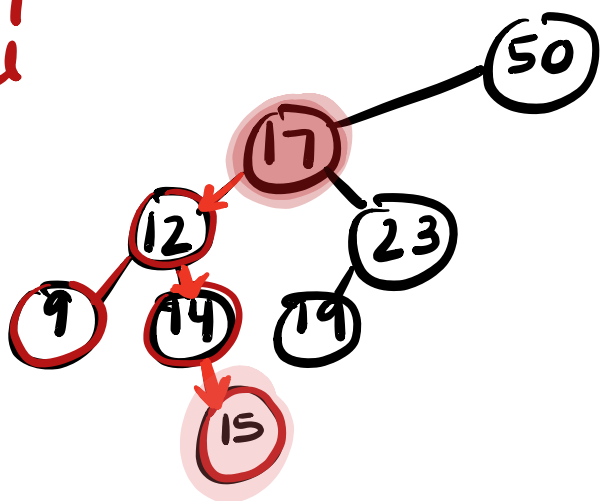
Ex) Predecessor of 17

= 14



Ex) Predecessor of 17 With a newly added Right leaf :

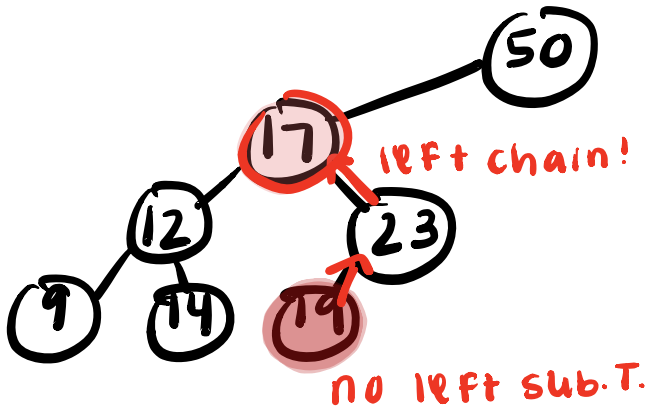
= 15



- If there is no left subtree, and the node is a left child, then you have to go up the tree and look for the chain that goes to the left

Ex) Predecessor of 19

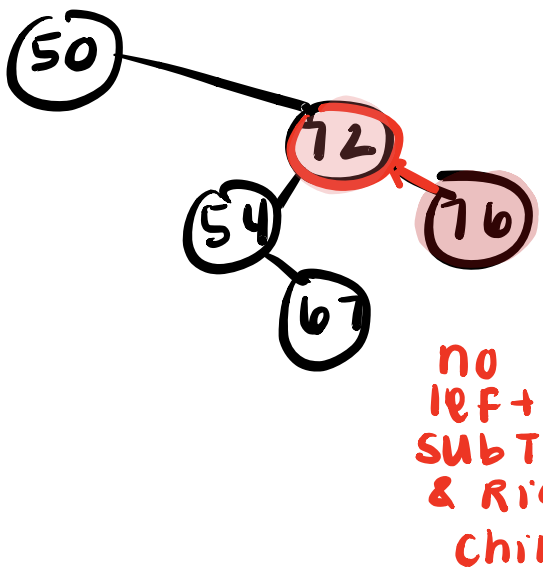
= 17

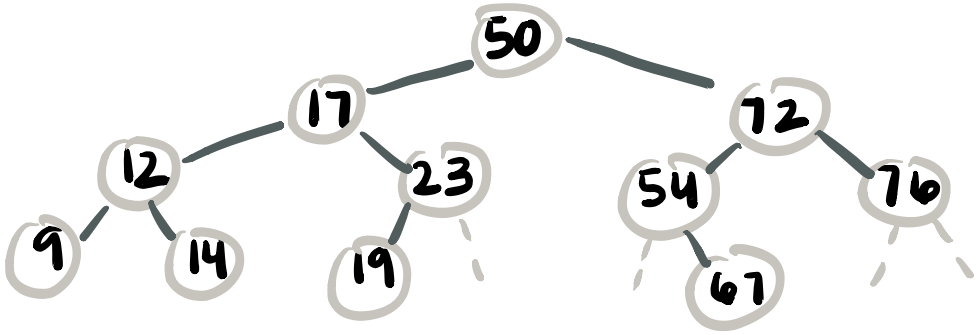


- If there is no left subtree, and the node is a Right child, then the predecessor will be the parent of the node.

Ex) Predecessor of 76

= 72





SUCCESSOR = smallest value greater than the node.

1) if has Right SubtRee, then it is the left most leaf. Ex: SUC(17) = 19

2) NO Right SubT., node is Right child, go up the tRee until you get to a Right Chain, then it is the value the Right chain points to
Ex: SUC(23) = 50

3) NO Right SubT., node is left child, it is the parent of that node. Ex: SUC(19) = 23

PREDECESSOR = largest value smaller than the node.

1) if has left subT., then it's the Right most leaf.
Ex: PRED(17) = 14

2) NO left subT., node is left child, go up the tree until left pointing chain, the it is that value.
Ex: PRED(19) = 17

3) NO left subT., node is Right child, it is the parent of that node.
Ex: PRED(76) = 72