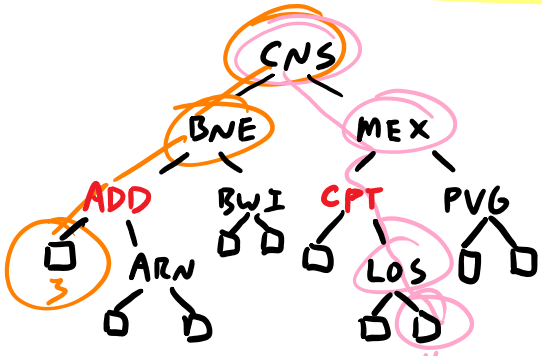
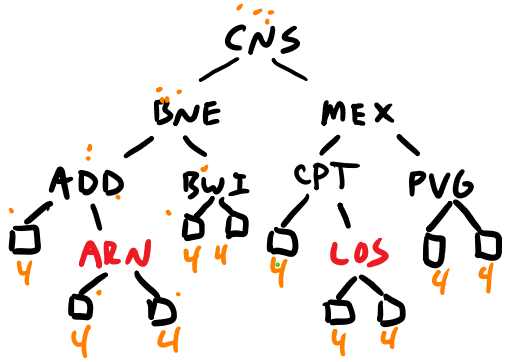
Red-Black Trees

Binary Search Trees such that

1) each node has a color: red or black
2) root is colored black
3) all leaves are colored black
4) children of red nodes are black
5) every path root → leaf has same # of black nodes

add dummy nodes s.t. all data nodes have 2 children





not a red-black tree
fails property 5

leaves in a non-empty full binary tree = # internal nodes (non-leaves) + 1 (data nodes)

dummy nodes

all nodes have 0 or 2 children

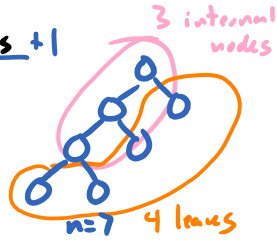
if height $O(\log_2$ total nodes$) = O(\log_2(2 \cdot$ # data nodes $+1)) = O(\log n)$

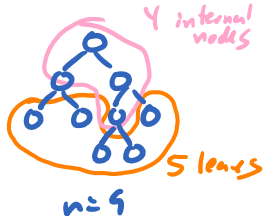
suppose # black nodes in path root → leaf is b

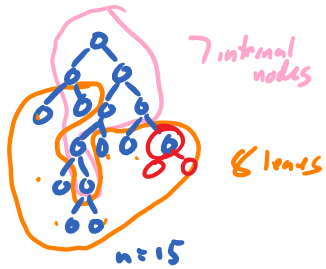
black height

shortest possible path has b nodes

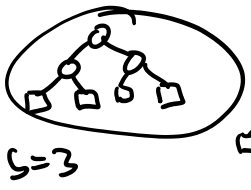
longest possible path has $2b-1$ nodes

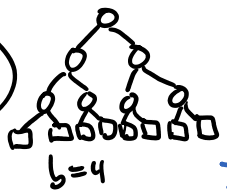






Red-black tree with black height b has $\geq 2^b - 1$ nodes





$1 + 2 + 4 + \dots + 2^{b-1} = 2^b - 1$

$n \geq 2^b - 1$

$n+1 \geq 2^b$

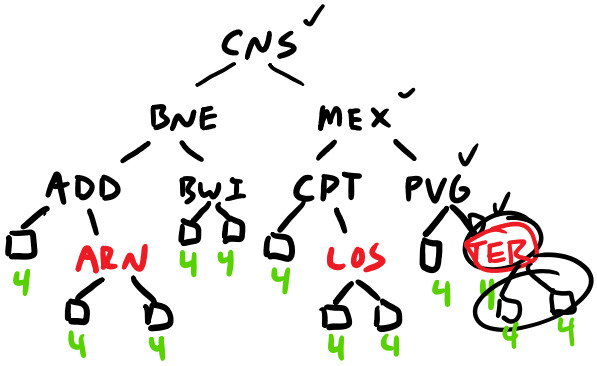
$\log_2 n+1 \geq b$

$b \leq \log_2 n+1$

$b \leq h \leq 2b-1 \leq 2\log_2(n+1)$

$h \leq n \leq 2h - 1 \leq 2\log_2(n+1)$

$O(\log n)$



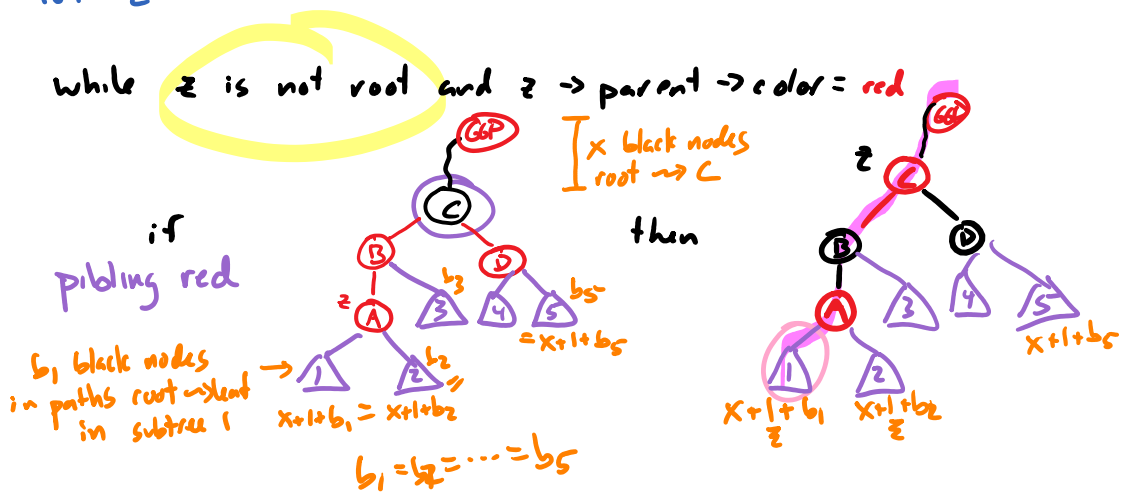
1) Do normal BST insert, color new node red (w/ black laws)

2) If · parent of new node exists and is black DONE

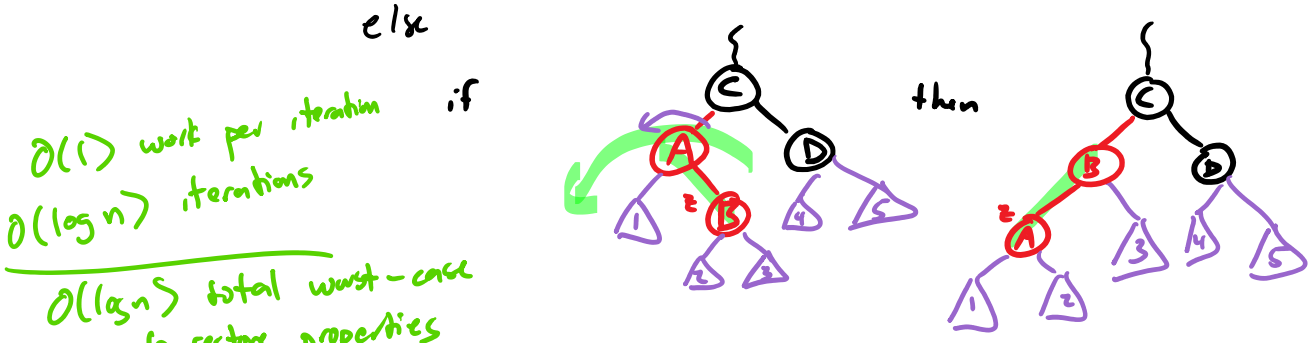
Else

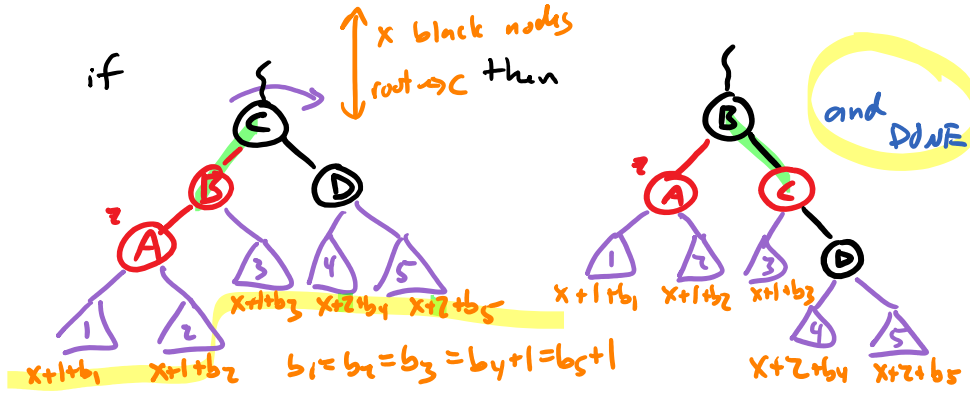
let z = new node

while z is not root and z -> parent -> color = red



else

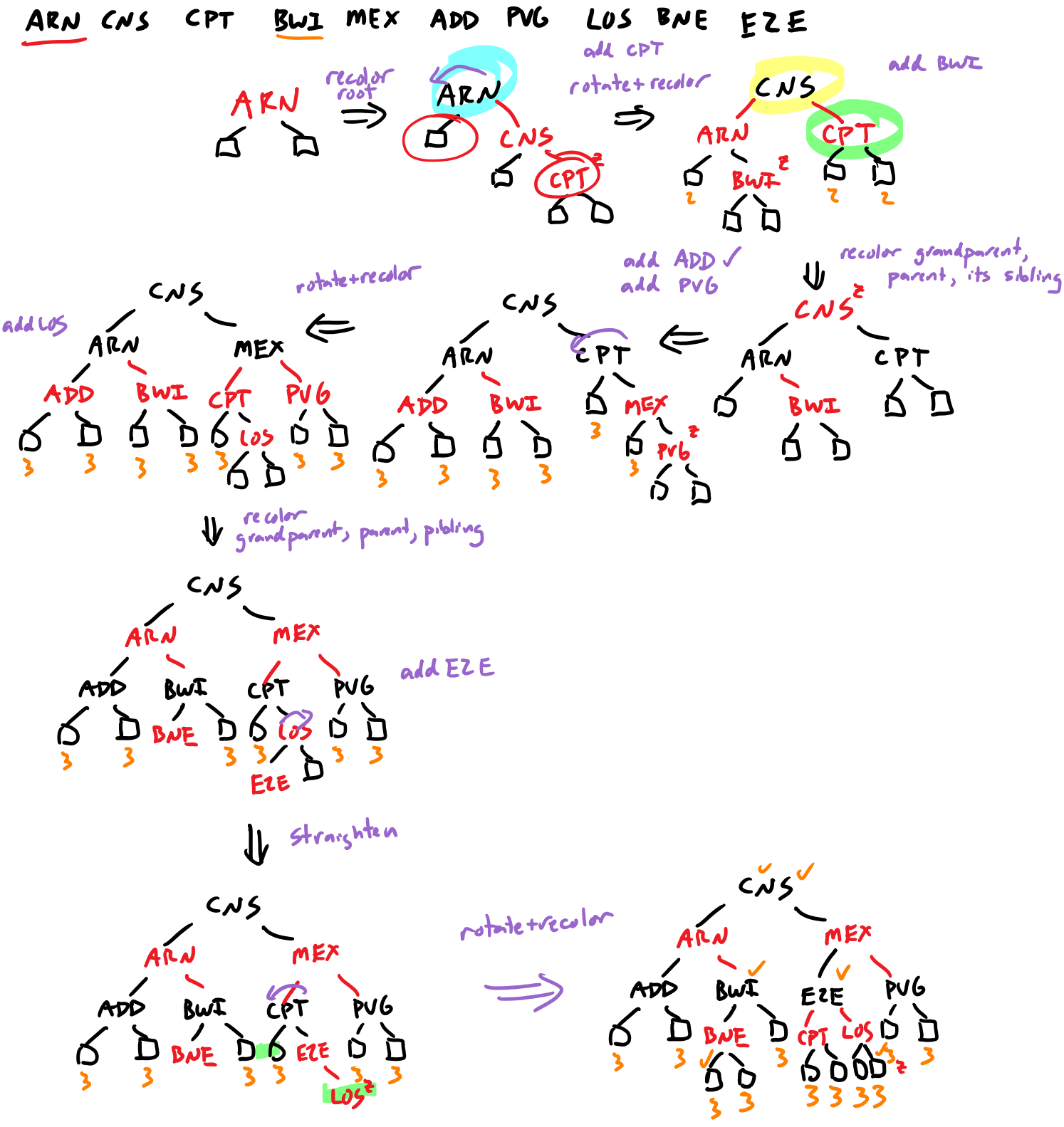




[other cases symmetric]

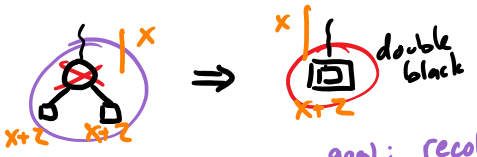
t → root → color = black

Example


ARN CNS CPT BWI MEX ADD PVG LOS BNE EZE
recolor root
add CPT
rotate+recolor
add BWI
add ADD ✓
add PVG
recolor grandparent, parent, its sibling
rotate+recolor
add LOS
recolor grandparent, parent, pibling
add EZE
Straighten
rotate+recolor

Red-Black Tree Delete

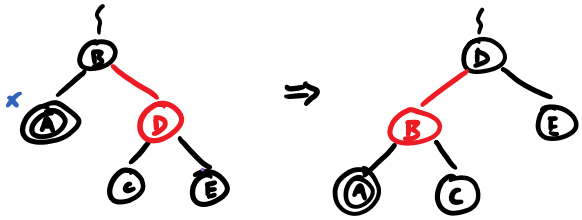
Do normal BST delete; if deleted node has 2 non-leaf children
then moved node takes deleted's color
do 1 non-leaf delete where moved node was

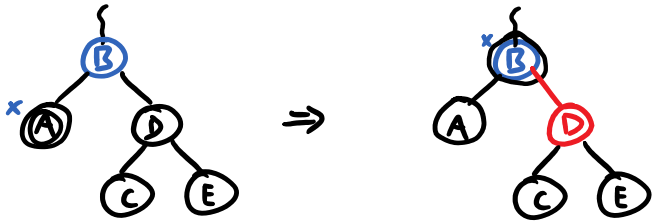


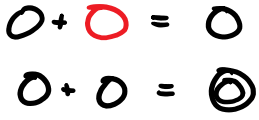
goal: recolor to eliminate double black or move double black to root

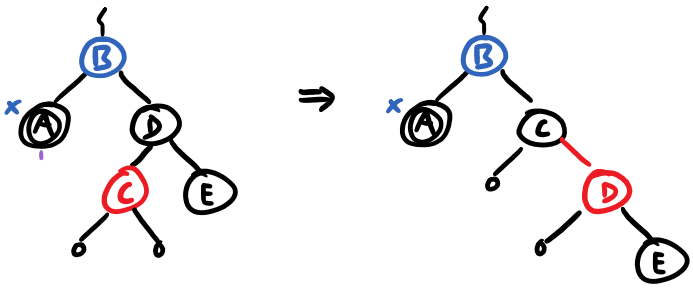
while x is doubly black

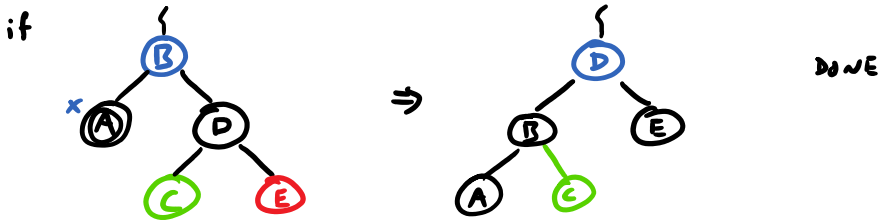
if  ⇒  DONE

if 

if 



else if 


if
B
A
D
C
E
D
B
E
A
C
Done