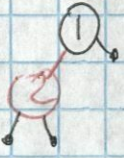
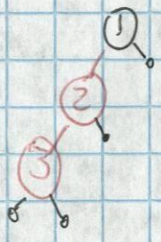
Gentry Atkins
CS5329 Fall 2019
Assignment #3

1) Insert 1

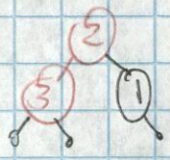
Insert 2



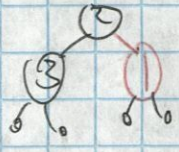
Insert 3



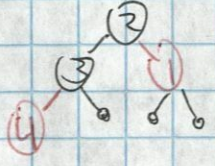
Rotate 2



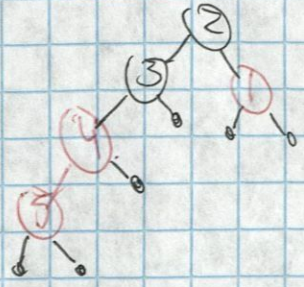
Recolor



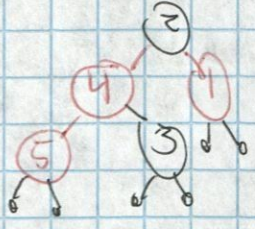
Insert 4



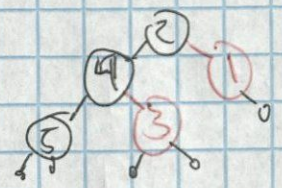
Insert 5



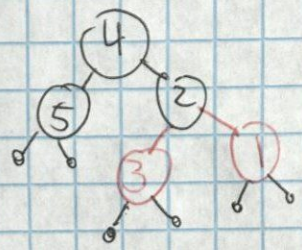
Rotate 4



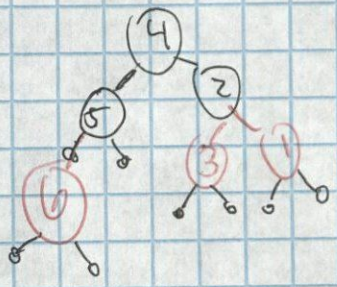
Recolor 5



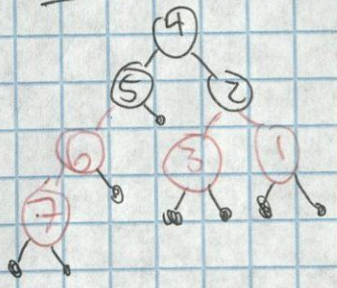
Rotate 4



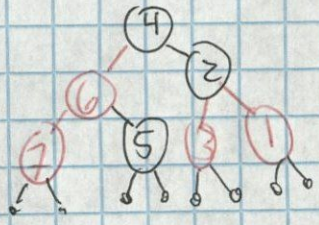
Insert 6



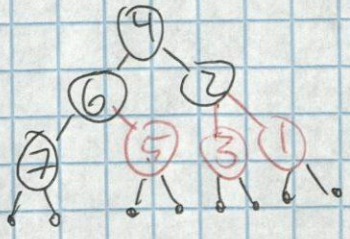
Insert 7



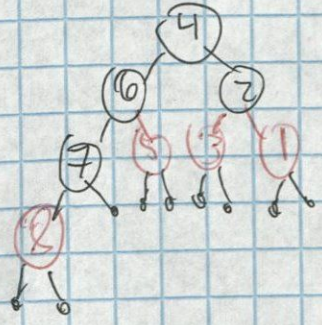
Rotate 6



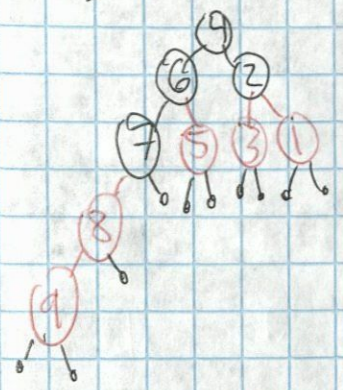
Recolor 7



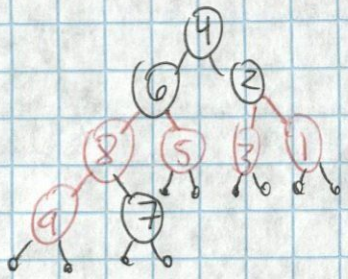
Insert 8



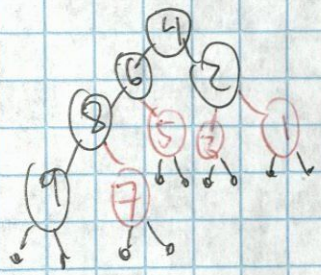
INSERT 9



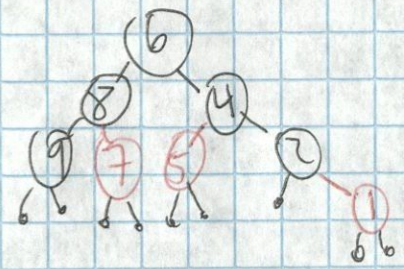
ROTATE 8



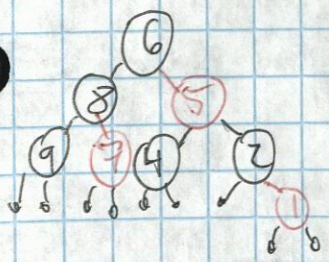
RECOLOR 9



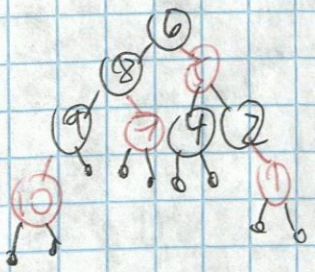
ROTATE 6



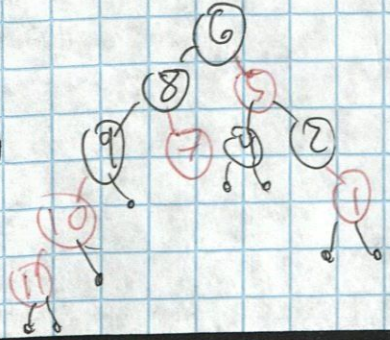
ROTATE 5



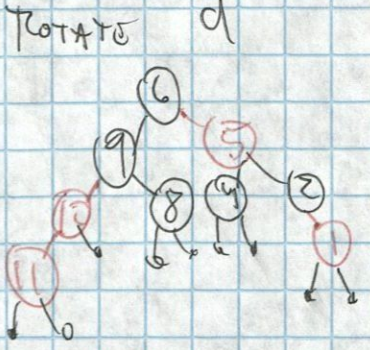
INSERT 10



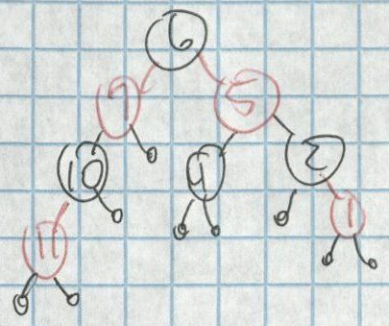
INSERT 11



ROTATE 9



RECOLOR 10



2) LARGEST # OF NODES : 2^{K}
SMALLEST # OF NODES : 2^{K-1}

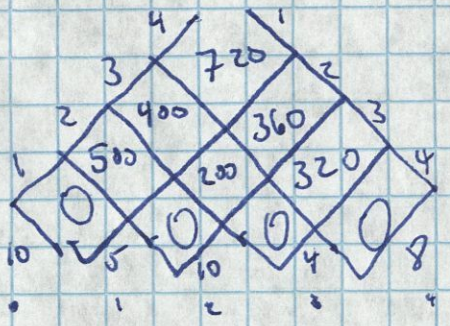
3) LIS (SEQ [], n, MAX, LIS[]) {
IF n == 1 RETURN 1;
IF n > MAX {
~~RETURN LIS~~
LIS. PUSH (n);
RETURN LIS(SEQ, n-1, ~~MAX~~, LIS);
}
ELSE
RETURN LIS (SEQ, n-1, MAX, LIS);

$T(n) = T(n-1) + n$
$\therefore T(n) = O(n^2)$

4) LCS (A[], B[], n, m, STRING*) {
IF (n == 0 OR m == 0)
RETURN 0
IF (A[n] == B[m])
STRING. PUSH (A[n])
RETURN MAX (LCS (A, B, n-1, m), LCS (A, B, n, m-1)) +1
RETURN MAX (LCS (A, B, n-1, m), LCS (A, B, n, m-1))
}

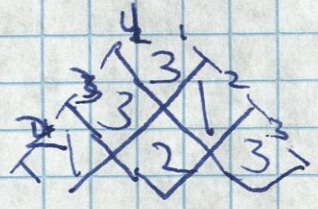
5) Optimal parenthization of $\langle 10, 5, 10, 4, 8 \rangle$

A_1 (10×5) A_2 (5×10), A_3 (10×4), A_4 (4×8)

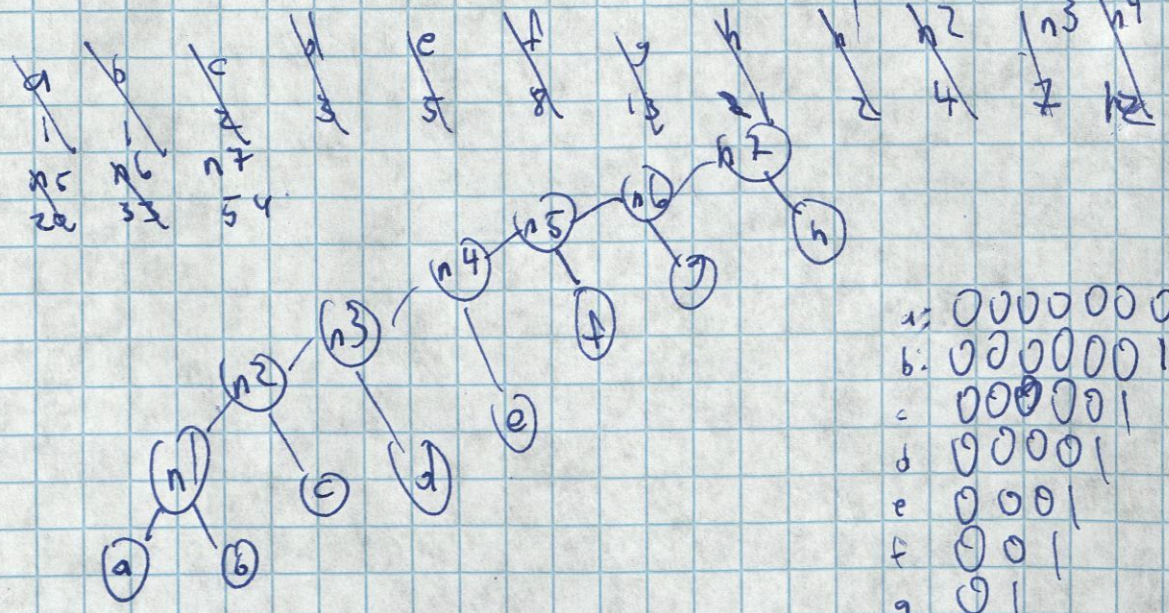


~~400~~
~~400~~
~~360~~
~~360~~
720

$((A_1 A_2) A_3)(A_4)$



6)



a: 0000000
b: 0000001
c: 000001
d 00001
e 0001
f 001
g 01
h 1