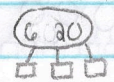
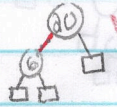
1. INSERTION 2-3 TREE R-B TREE

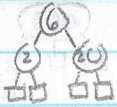
(20)

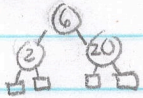
(6)



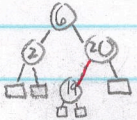


(2)

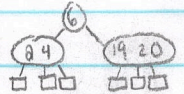


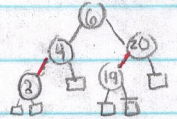


(19)

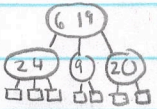


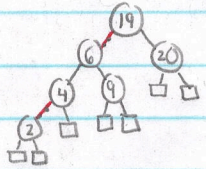
(4)





(9)



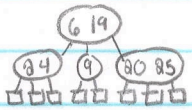
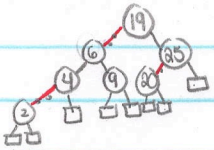
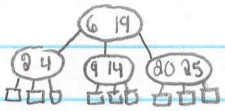
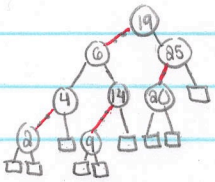
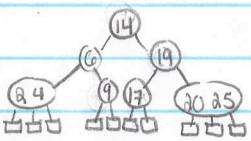
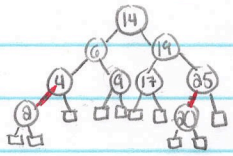
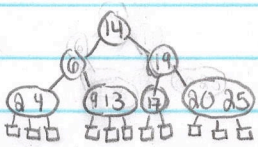
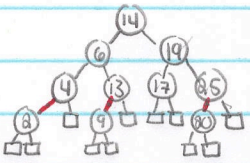


Continues →

CSC 225 Assignment #5

Cont. →

1. INSERTION

	2-3 TREE	R-B TREE
(25)		
(14)		
(17)		
(13)		

2. Algorithm: preOrder (Tree T, Node V)

p = primary operation

C = comparison

```
if T.isInternal(v) then
    Process Node (v)
    preorder (T, T.leftChild(v))
    preorder (T, T.rightChild(v))
    end
end
```

$1C + 1p + \{$

Cp

$T(n_L) = T(n)$ since no internal nodes to right

$T(n_r) = T(0)$ since right child is external node

$\}$

$$T(n) = \begin{cases} 2 & n = 0 \\ 2 + C + T(n) + T(0) & n \geq 1 \end{cases}$$

$$T(n) = 4 + C + T(n_-) = (4 + C + (4 + C + T(n_- - 1))) = 2(4) + 2C + T(n_- - 1)$$

Continues →

Cont. →

2. $T(n) = (4+c+(4+c+(4+c+T(n-2)))) = 3(4)+3c+T(n-2)$

$= 4i + ci + T(n-i-1)$ let $n-1=i$

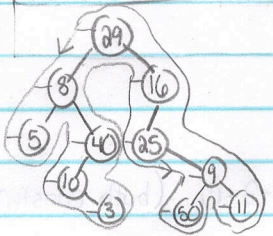
$= 4(n-1) + c(n-1) + T(n-n+1-1)$

$= 4n-4+cn-c+T(0)$

$= n(4+c)-2-c$ let $C=4+c$ and $K=-2-c$ (both constants)

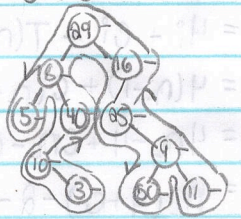
$= Cn+K \in O(n)$

3. a) Pre order:



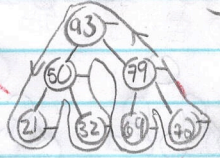
29	8	5	40	10	3	16	25	9	50	11

Post-order:



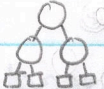
5	3	10	40	8	50	11	9	25	16	29

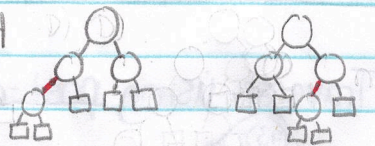
b) Traverse using post-order traversal

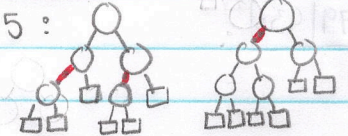


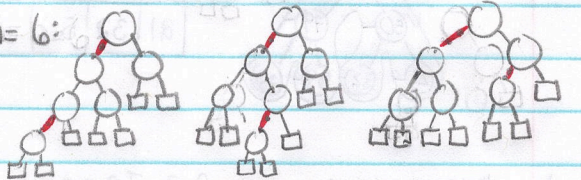
21	32	50	69	72	79	93

4. n=2 : 

n=3 : 

n=4 

n=5 : 

n=6 : 

n=7 :