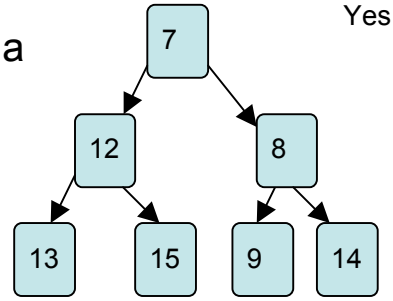
Section Worksheet #4 Solutions

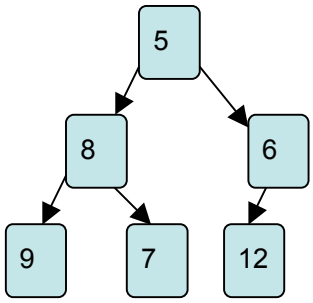
1.

1a



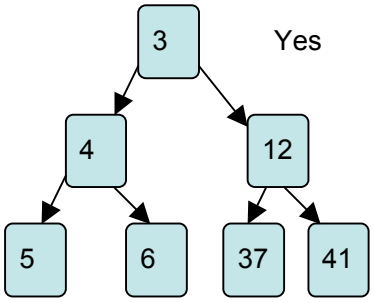
Yes

1b



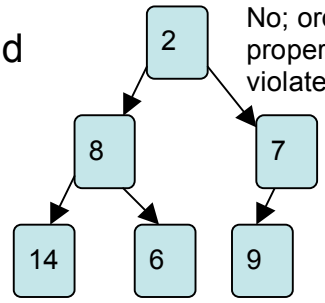
No; ordering property violated

1c



Yes

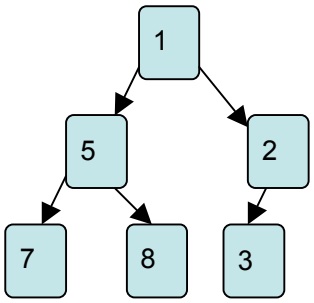
1d



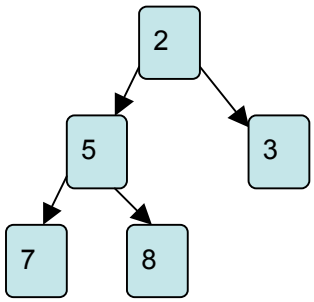
No; ordering property violated

e. It would have a gap in the array; to be complete the array should be populated from index 1 to n, and nowhere else (except 0, which may store the size or something similar).

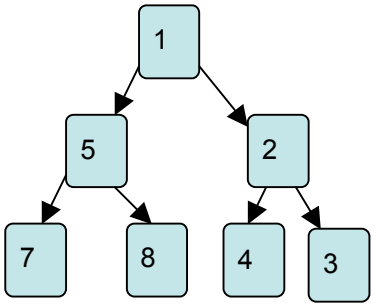
2. a)



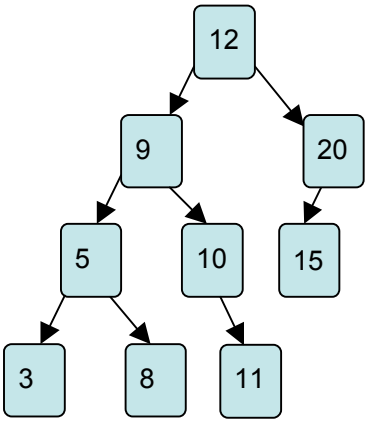
2. b)



2. c)



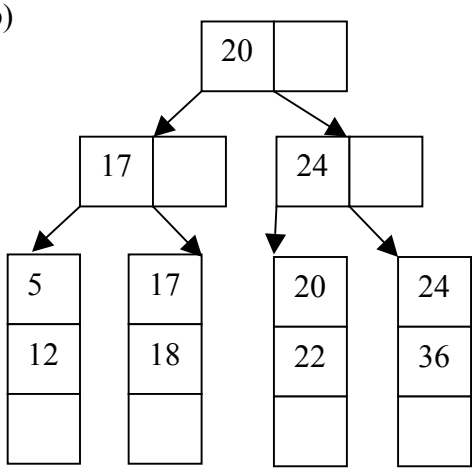
3. a)



b) Adding a node as either a left child of node 15 or node 3 would cause a left-left imbalance

4. a) Each node can have at most M children, and must have a minimum of M/2 children, and each leaf can have at most L data items, and at least L/2 data items. (rounding-up) So, a tree with M=32 and L=16 must have 16-32 children at each internal node, and must have 8-16 items at each leaf. (excepting the first 7 insertions)

4. b)



4. c)

18
20
24

5. a)

0	14
1	
2	
3	24=>17=>10
4	4
5	12
6	13

5. b)

0	14
1	13
2	24
3	10
4	17
5	4
6	12

5. c) 14, 10, 17, 4, 12, 13, 24

0	14
1	13
2	
3	10
4	17
5	4
6	12

We can never place the 24 because the index loops between 4,0,5,5,0,4,3 and never hits 2.