Eli Sobylak
27-nov-15

Set 4: 4.1, 4.2, 4.7, 4.9, 4.11, 4.19, 4.26, 4.27, 4.31, 4.43, 4.45

4.1:

a. The root node in this figure is A
b. the nodes which are leaves are G,H,I,L,M, and K

4.2:

a. Parent node of each
    - A: []
    - B: [A]
    - C: [A]
    - D: [B]
    - E: [B]
    - F: [C]
    - G: [D]
    - H: [D]
    - I: [E]
    - J: [E]
    - K: [F]
    - L: [J]
    - M: [J]

b. Child of each node
    - A: [B,C]
    - B: [D,E]
    - C: [F]
    - D: [G,H]
    - E: [I,J]
    - F: [K]
    - G: []
    - H: []
    - I: []
    - J: [L,M]
    - K: []
    - L: []
    - M: []

c. Siblings of each node
    - A: []
    - B: [C]
    - C: [B]
    - D: [E]
    - E: [D]
    - F: []
    - G: [H]

H: [G]
I: [J]
J: [I]
K: []
L: [M]
M: [L]

d. Depth of each node

A: [0]
B: [1]
C: [1]
D: [2]
E: [2]
F: [2]
G: [3]
H: [3]
I: [3]
J: [3]
K: [3]
L: [4]
M: [4]

e. Height of each node

A: [4]
B: [3]
C: [3]
D: [2]
E: [2]
F: [2]
G: [1]
H: [1]
I: [1]
J: [1]
K: [1]
L: [0]
M: [0]

4.26:

```
static Position DoubleRotateToLeft(AvlNode * & k3) {
        Position k1,k2;
        k1 = k3 -> left;
        k2 = k1 -> right;
        k1 -> right = k2 -> left;
        k3 -> left = k2 -> left;
        k2 -> left = k1;
        k2 -> right = k3;
        k1 -> height = max(hiehgt(k1 -> left), hieght(k1 -> right)) + 1;
        k3 -> height = max(height(k3 -> left), height(k3 -> right)) + 1;
```

```
        k2 -> height = max(k1 -> height, k3 -> hight) + 1;
        return k2
}
```