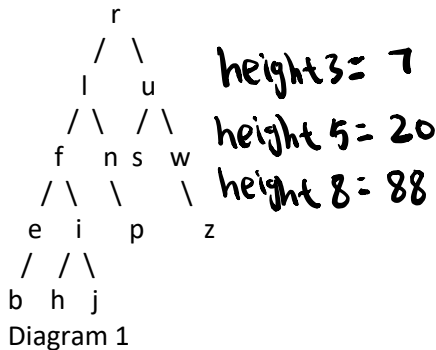


Part B

- What is the minimum number of node in an AVL tree with height of 3, height 5 and height 8?
Show the code below and write down the result.

Consider following diagram showing the state of an AVL tree



- Write the inorder traversal of the tree.

b, e, f, h, i, j, l, n, p, r, s, u, w, z

```

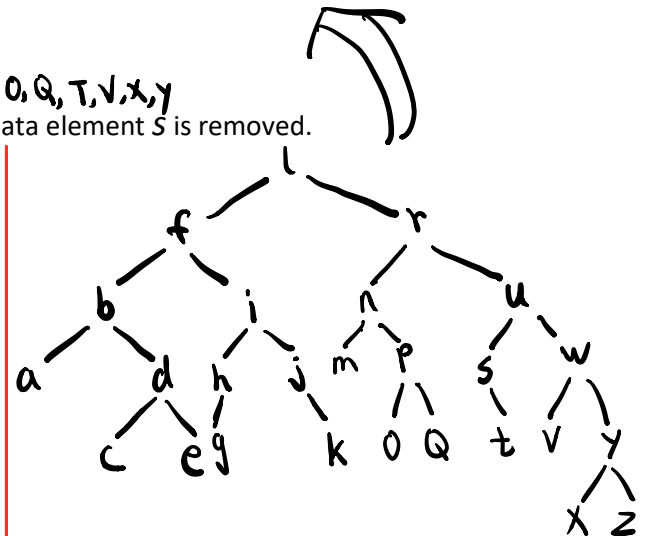
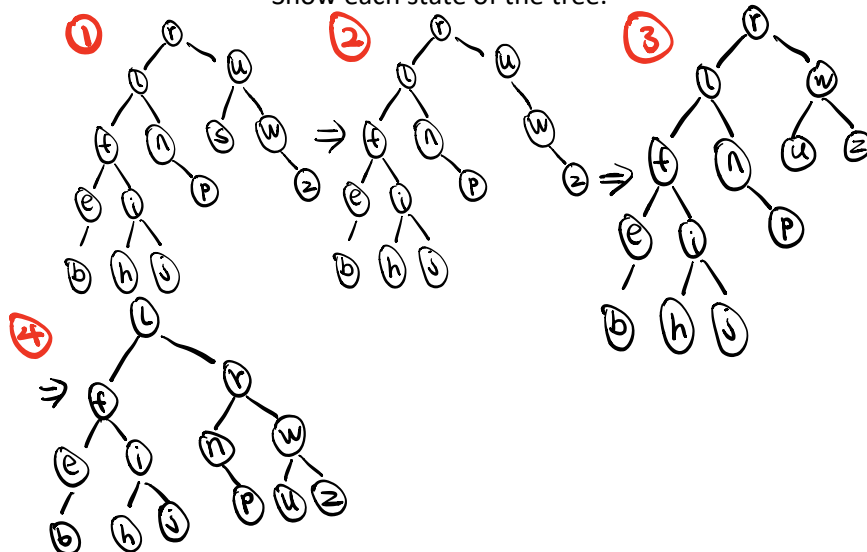
Public class AVL {
    Public static void main (String args[]) {
        int h1 = 3;
        int h2 = 5;
        int h3 = 8;
        System.out.println ( minimum (h1));
        System.out.println ( minimum (h2));
        System.out.println ( minimum (h3));
    }
    Public static int minimum (int height) {
        if (height == 0) return 1;
        if (height == 1) return 2;
        return (1 + minimum (height - 1) +
                minimum (height - 2));
    }
}
    
```

- List all single lower case letters whose insertion into an AVL Tree represented by Diagram 1 above would require a rebalance of the tree. (Insert a – z in alphabetic order. Remember BST contains no duplicate data).

add: a, c, d, h, k, m, o, q, t, v, x, y

- Show how the tree in Diagram 1 is changed when the data element S is removed. Show each state of the tree.

remove: s



cause-rebalance: a, c, d, h, y

- What is the worse-case runtime of deleting an element from an AVL tree? What about an unbalanced binary search tree?

AVL worst case: $O(\log_2 n)$

unbalanced binary search tree $O(n)$