

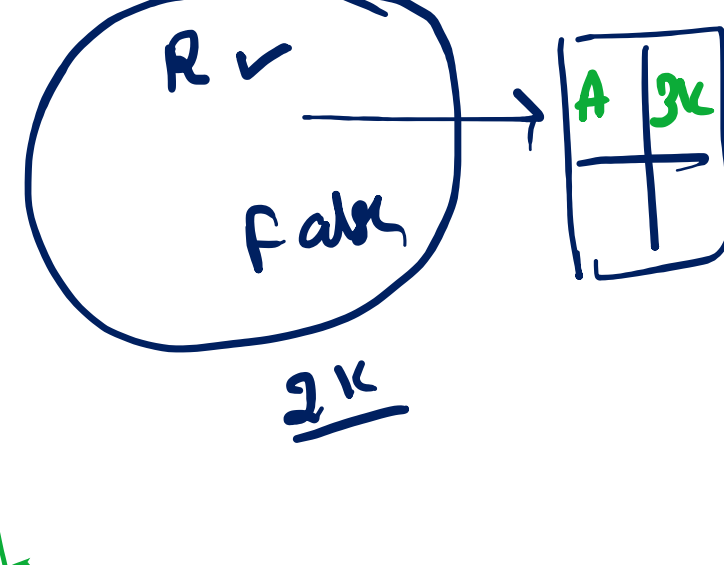
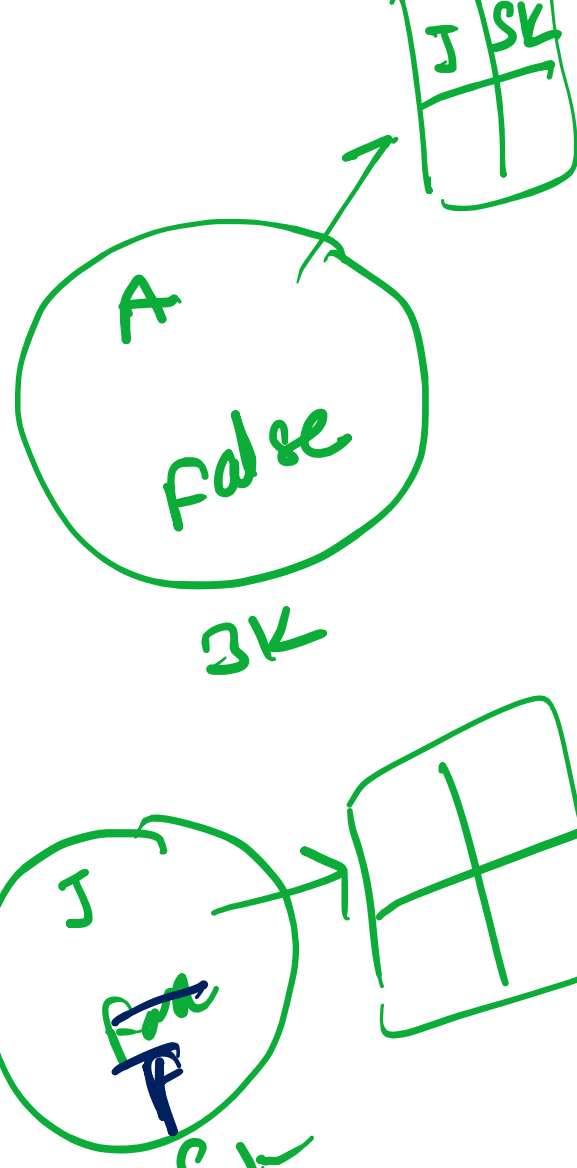
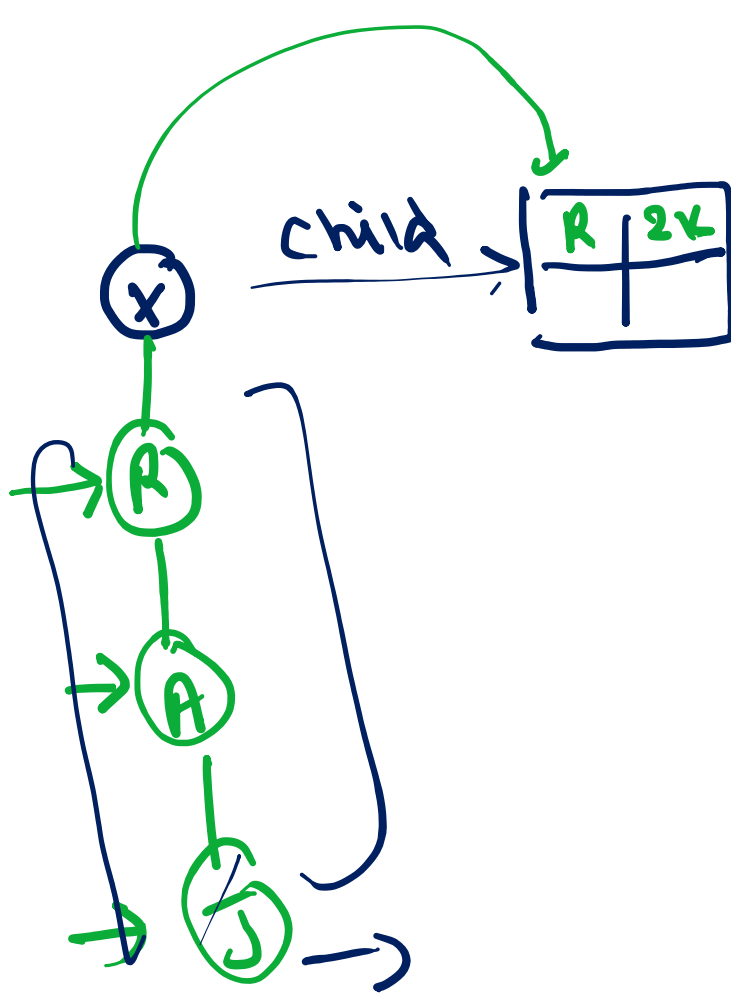
RIA/m/g

RIA/m/g

Node class {
char data
HashMap<char, Node> child
boolean isTerminal

```
class Node {  
    Character ch;  
    HashMap<Character, Node> child = new HashMap<>();  
    boolean isTerminal; // false  
}  
  
private Node root;  
  
public Trie() {  
    // TODO Auto-generated constructor stub  
    Node nn = new Node();  
    nn.ch = '*';  
    root = nn;  
}
```

RAJ



```
public void insert(String word) {  
    Node curr = root;  
    for (int i = 0; i < word.length(); i++) {  
        char ch = word.charAt(i);  
        if (curr.child.containsKey(ch)) {  
            curr = curr.child.get(ch);  
        } else {  
            Node nn = new Node();  
            nn.ch = ch;  
            curr.child.put(ch, nn);  
            curr = nn;  
        }  
    }  
}
```

4
add hack
add hackerrank
find hac
find hak

Sample Output

