

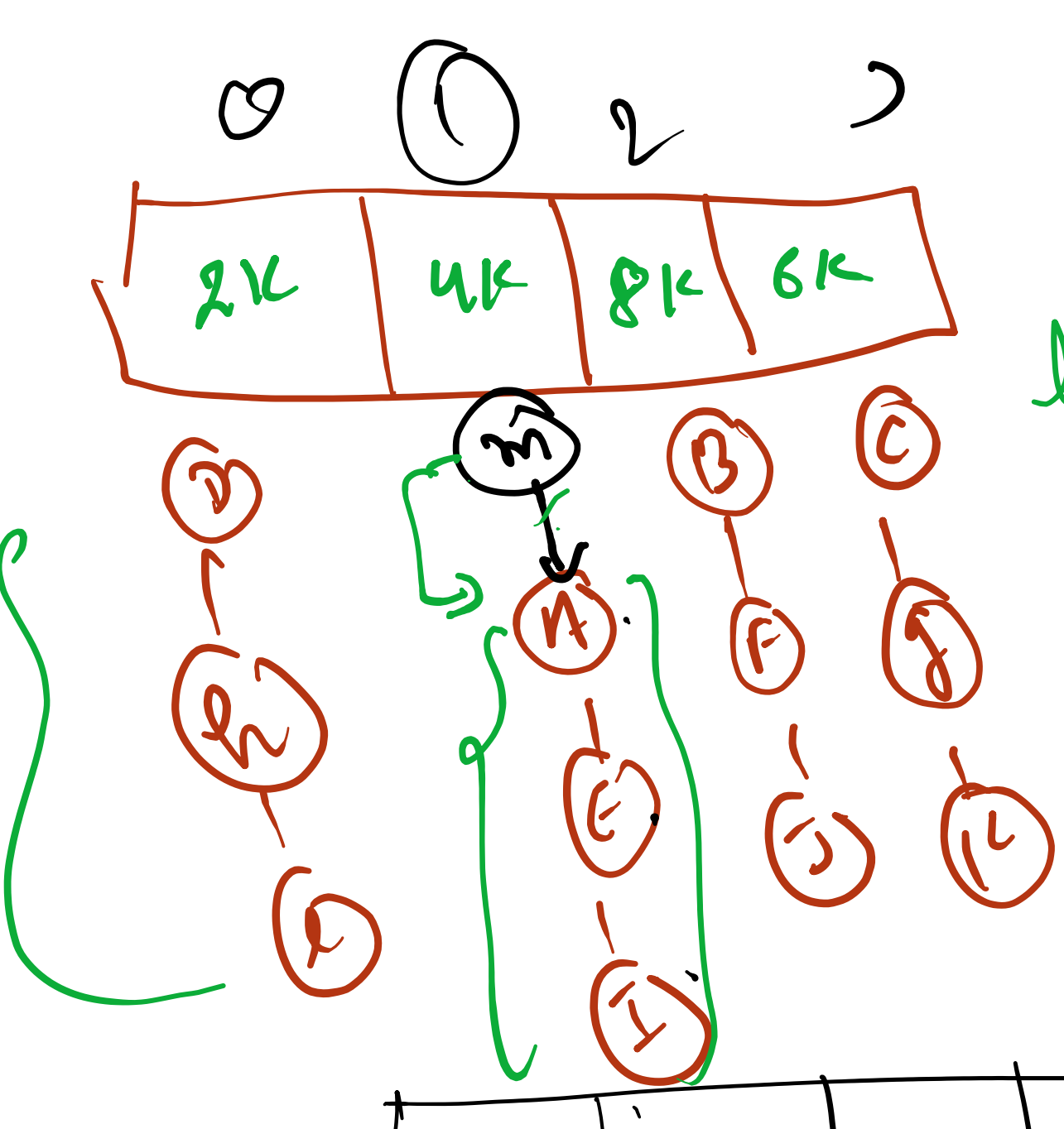
① ← A, 57
C, 77
D, 87
B, 56

hash fun
A 57 → 1
C 77 → 3
D 87 → 2
B 56 → 4

65 % 4 = 1
67 % 4 = 3
68 % 4 = 0
66 % 4 = 2
69 % 4 = 1

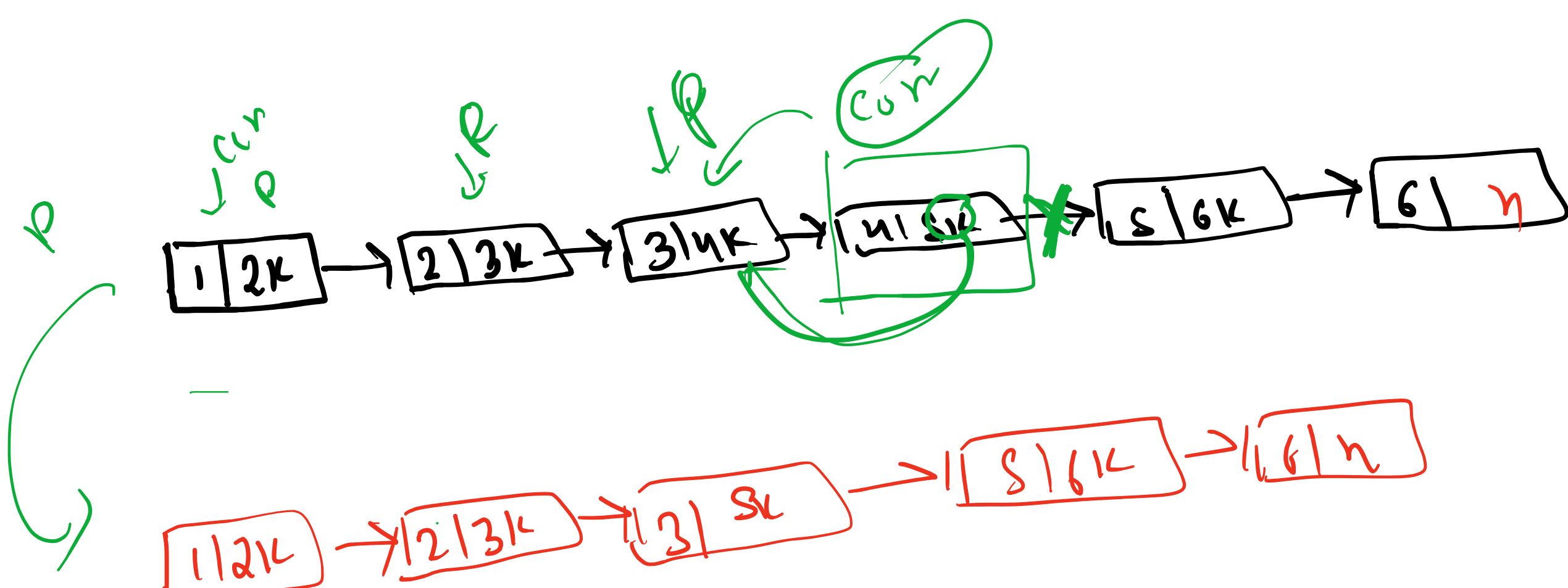
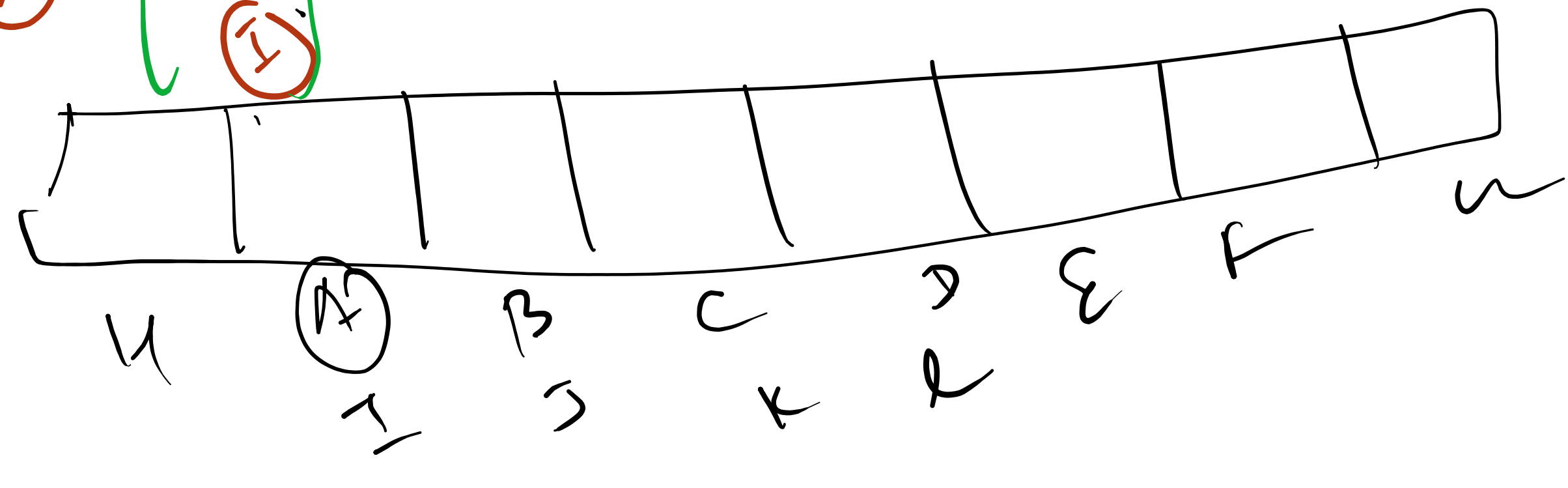
For C Node nn; old null
while (nn != null) {
put C nn-key, nn-value
nn = nn.next
}

```
public void put(K key, V value) {  
    int idx = hashfun(key);  
    Node temp = ll.get(idx);  
    while (temp != null) {  
        if (temp.key.equals(key)) {  
            temp.value = value;  
            return;  
        }  
        temp = temp.next;  
    }  
    temp.next = new Node(key, value);  
}
```



hash fun

THF = 2.0
LF = 12/4 = 3.0
2.1

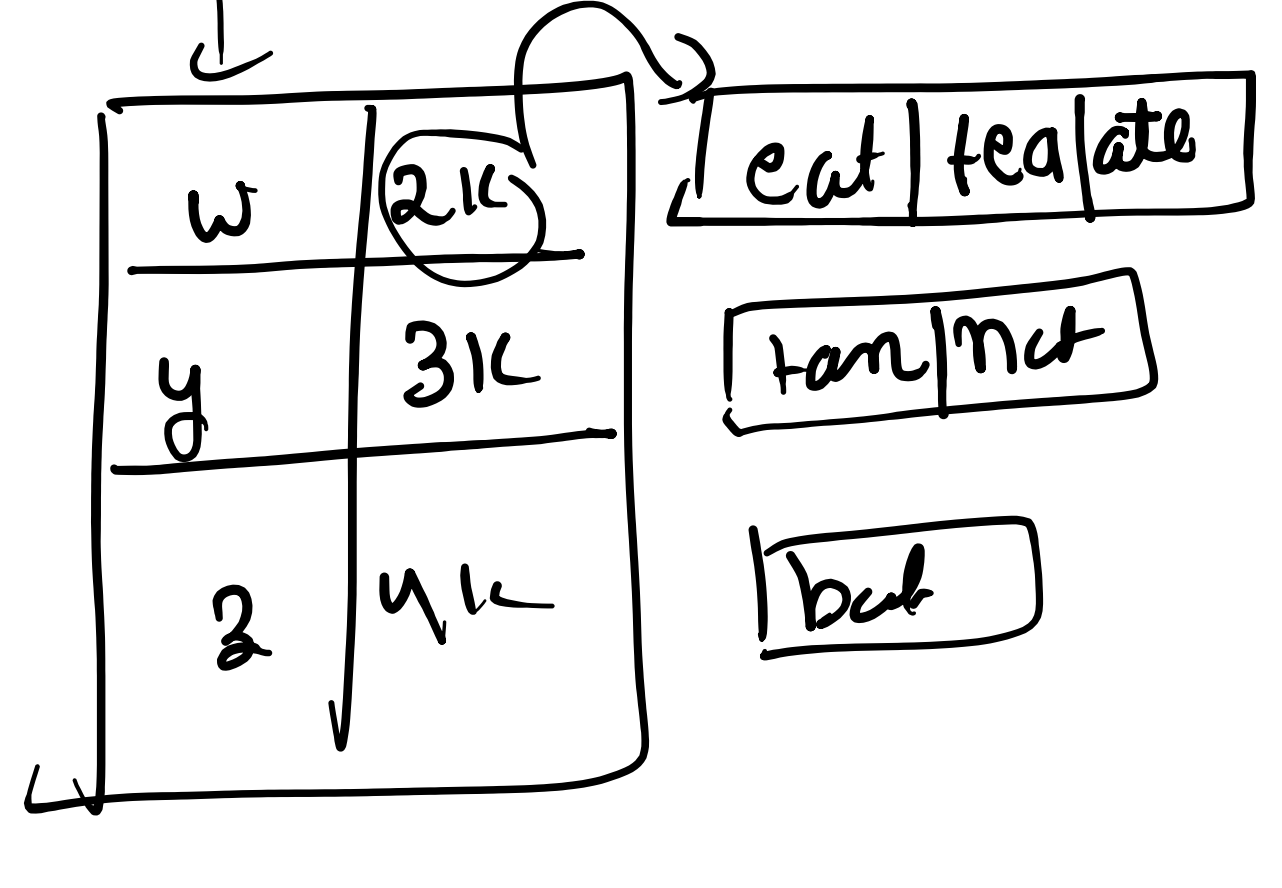
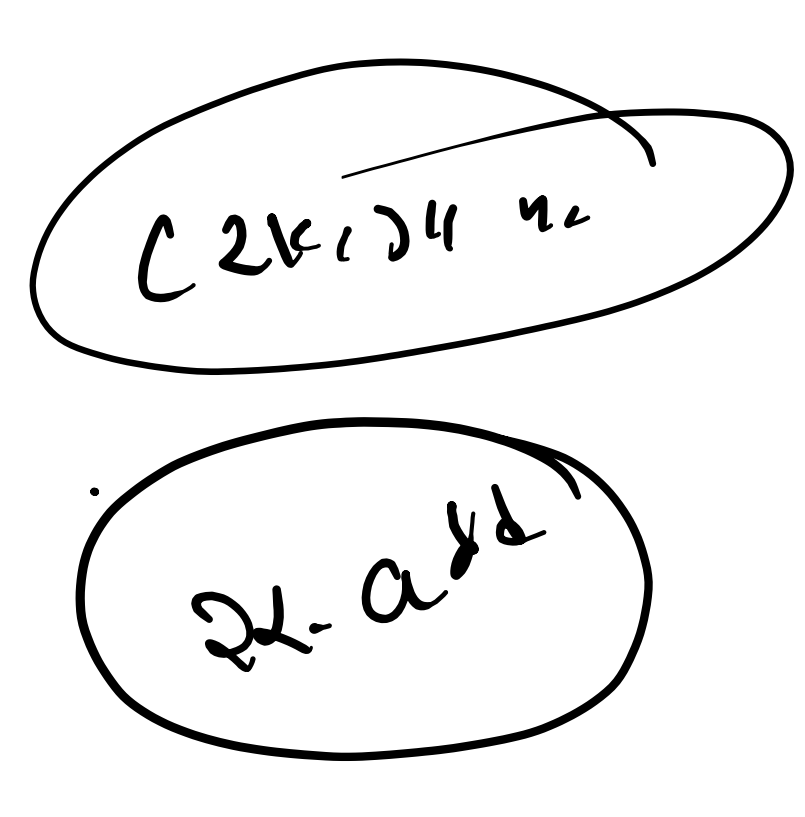
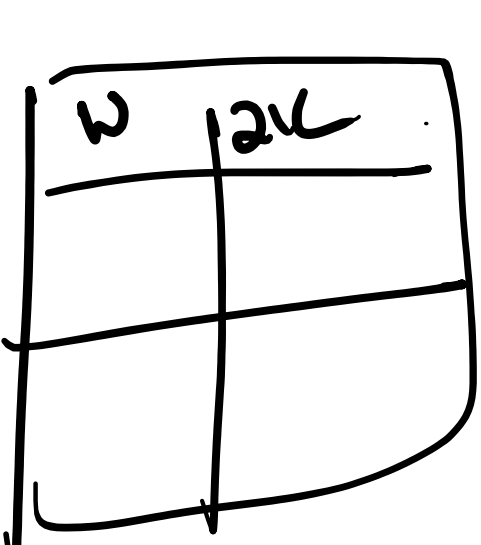


[eat, tea, ate] [nat, tan] [bat]

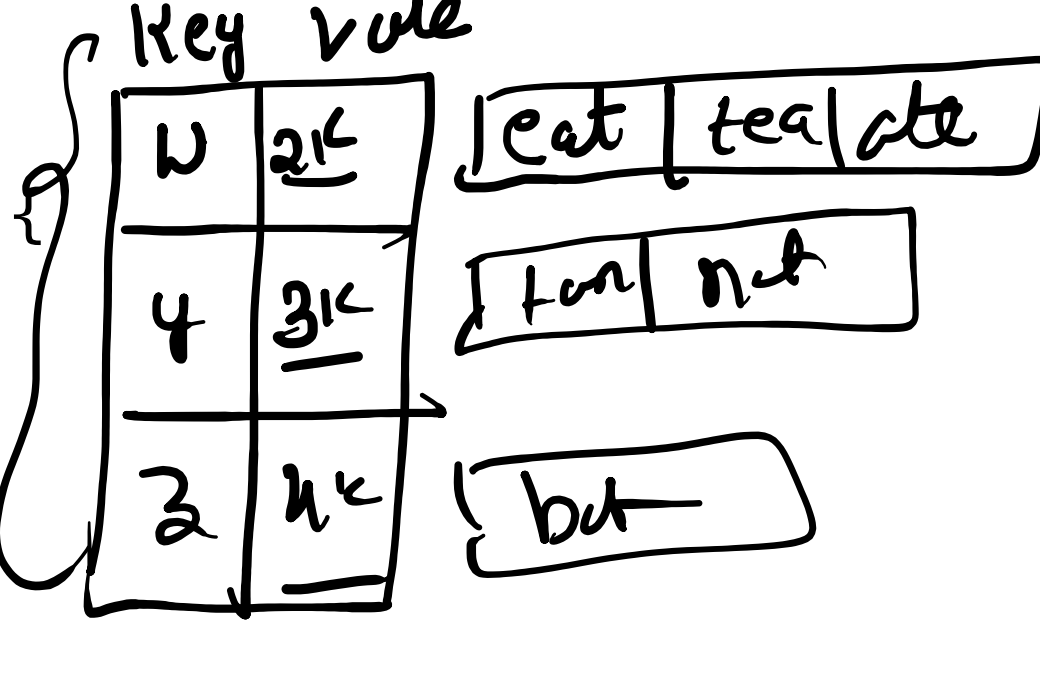
get get ant act ant abt

Input: strs = ["eat", "tea", "tan", "ate", "nat", "bat"]

ac = 97 + 99
bb = 98 + 98

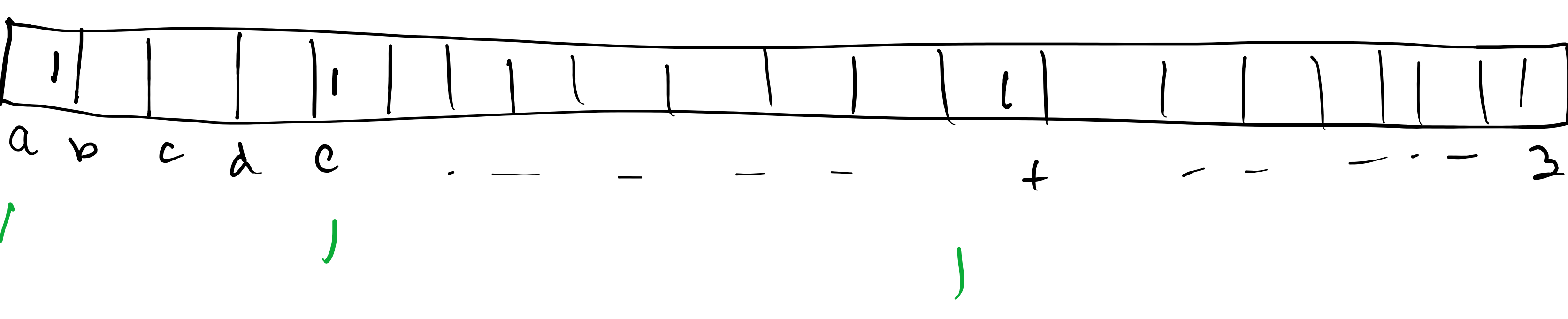


```
public static List<List<String>> groupAnagrams(String[] strs) {  
    HashMap<String, List<String>> map = new HashMap<>();  
    for (int i = 0; i < strs.length; i++) {  
        String key = getKey(strs[i]);  
        if (!map.containsKey(key)) {  
            map.put(key, new ArrayList<>());  
        }  
        map.get(key).add(strs[i]);  
    }  
}
```

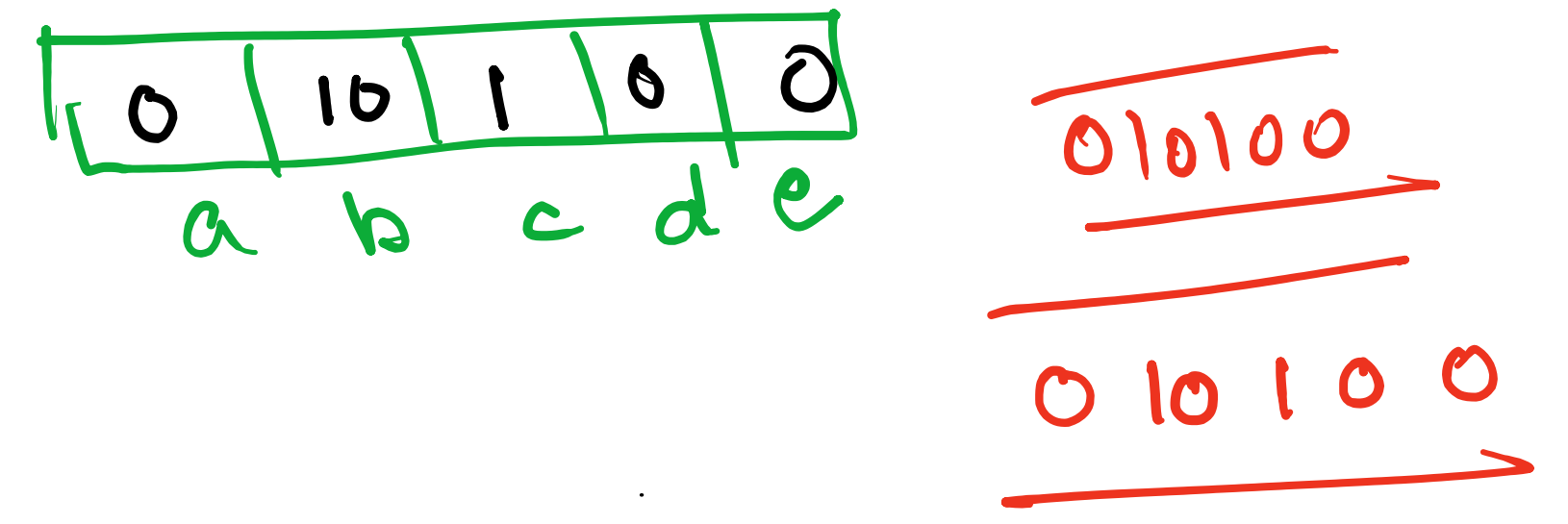
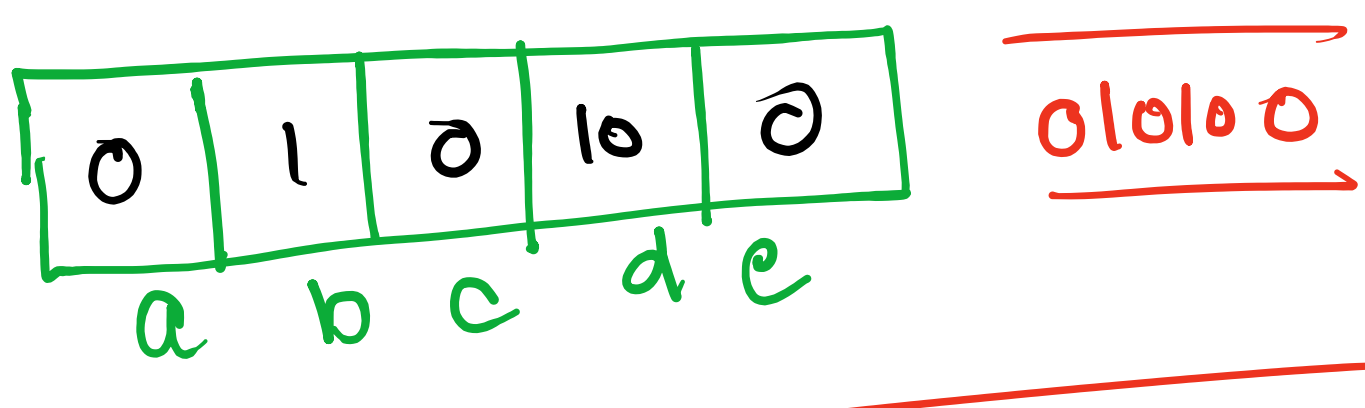


Input: strs = ["eat", "tea", "tan", "ate", "nat", "bat"]

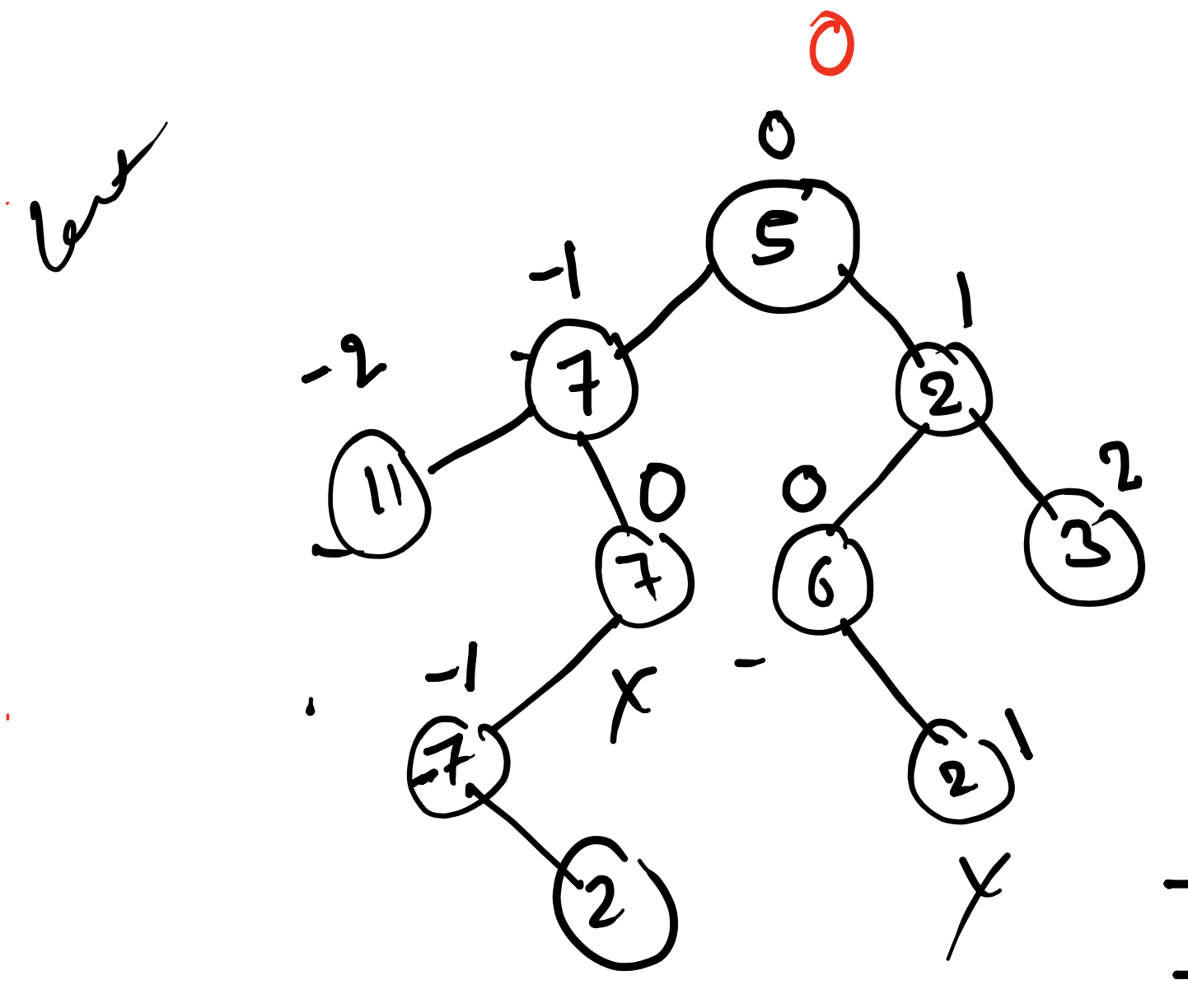
1000100...10...



["b d d d d d d d d d", "b b b b b b b b b c"]



010100
010100



[11, 7, 5, 2, 3]

