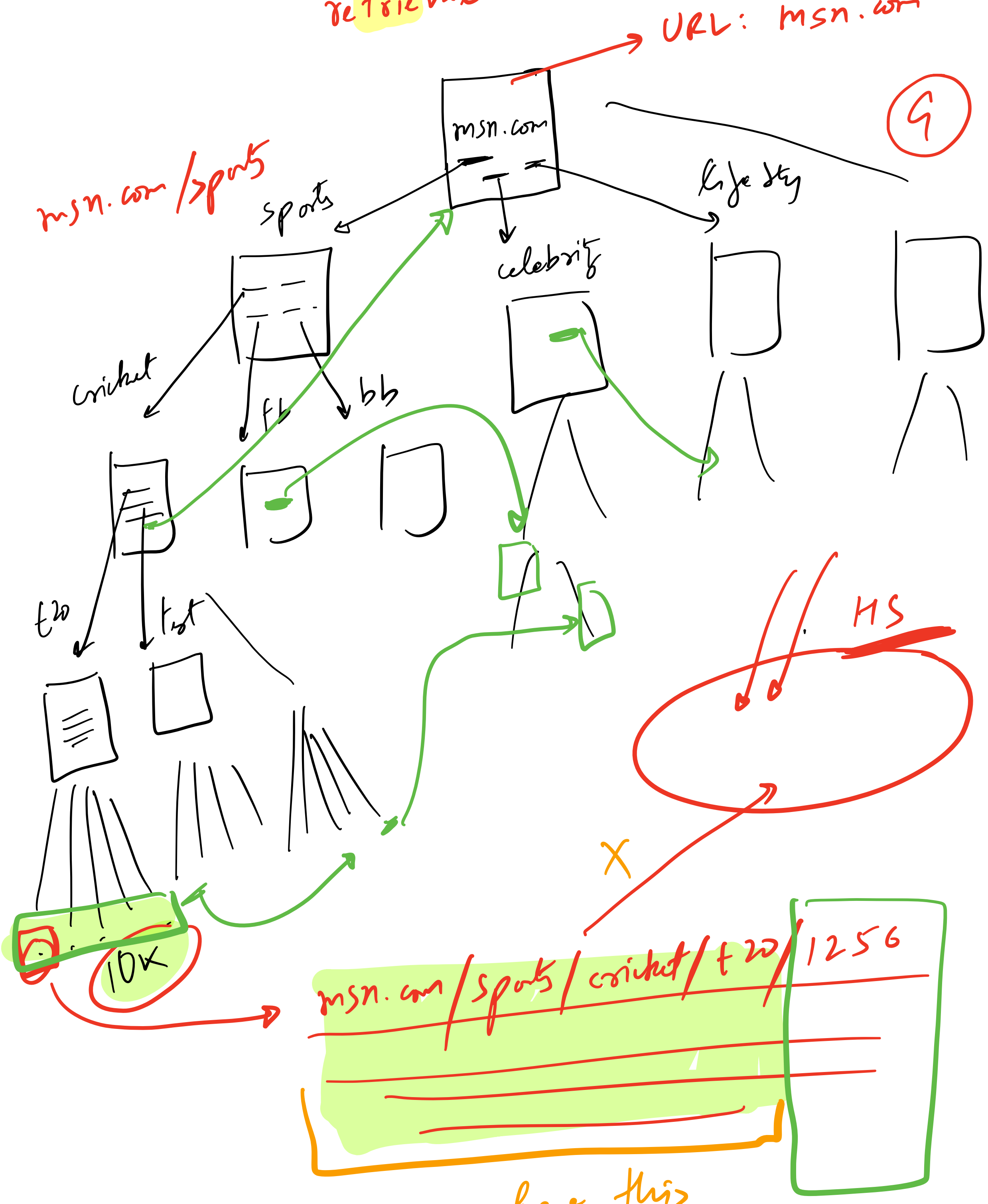
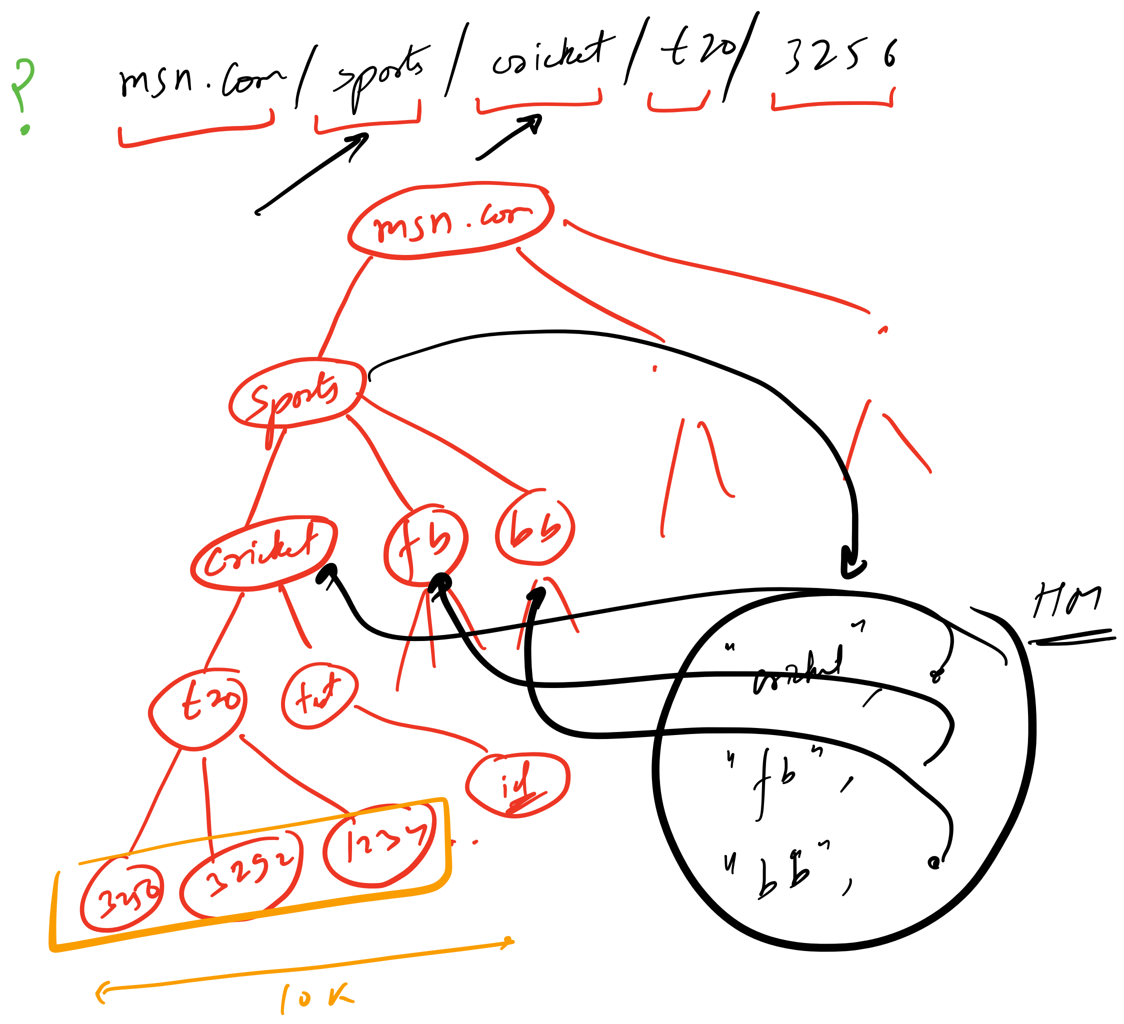
Trie $\rightarrow$: Tree

retrieval

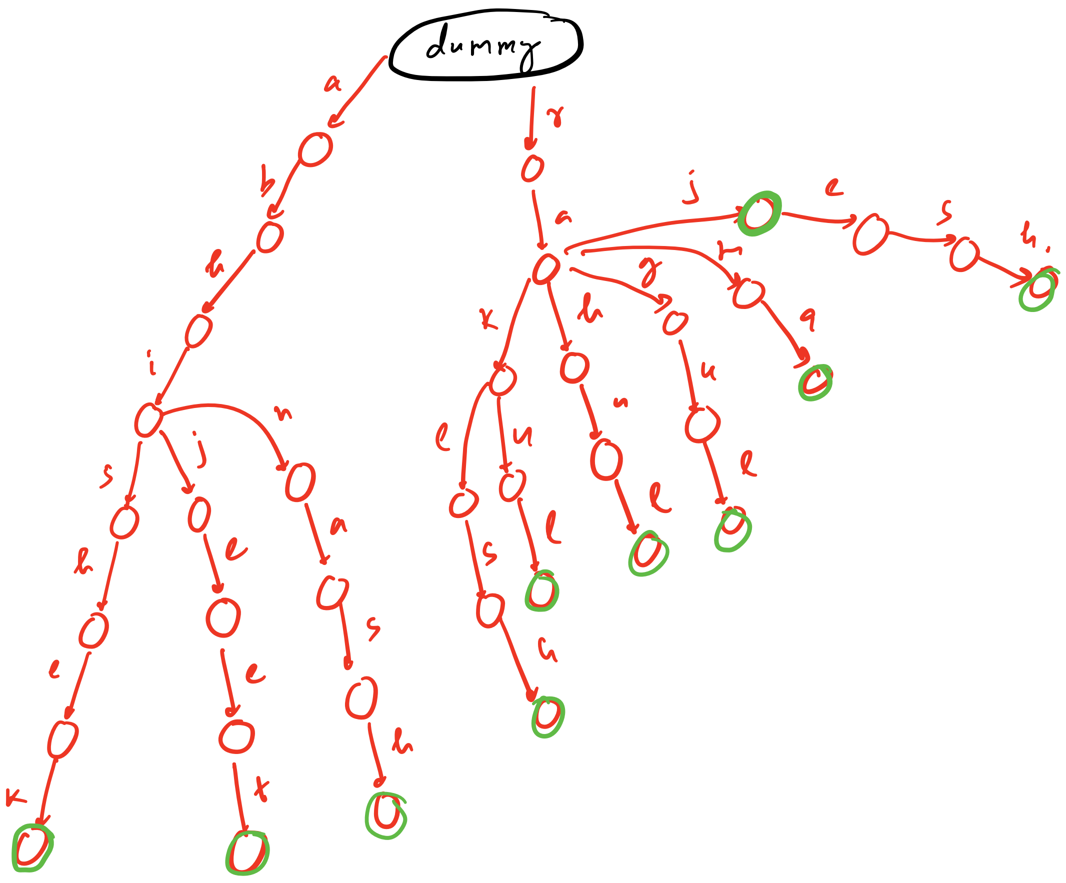


10k pages have this URL in common!


?
msn.com / sports / cricket / t20 / 3256
msn.com
Sports
Cricket
fb
bb
t20
tst
id
3256
3292
1234
10K
HM
"cricket",
"fb",
"bb",

Trie on characters → : Prefix DS

abhishek
abhijeet
abhinash
rahul
ragul
rakul
rakesh
rajesh
rama
raj



Google Search

ind vs pak
ng

phone contacts →

8088

IDE →

fo

Auto suggestion

ba ck
ng
d

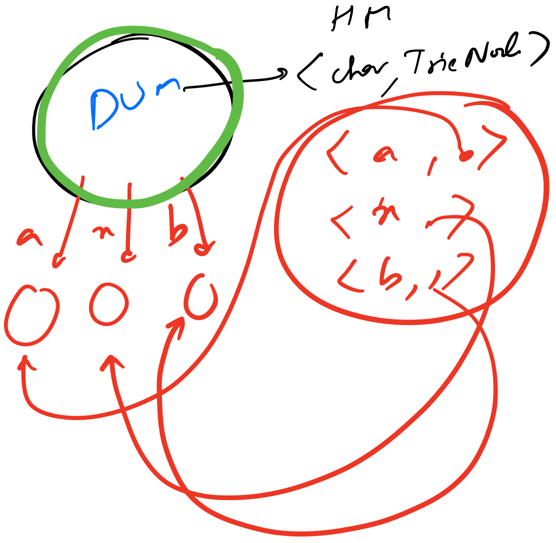
tries are:

→ Memory friendly when data has common prefixes

→ Searching is faster!

```
TrieNode {
    HashMap <char, TrieNode > child;
    bool isEnd;

}
```



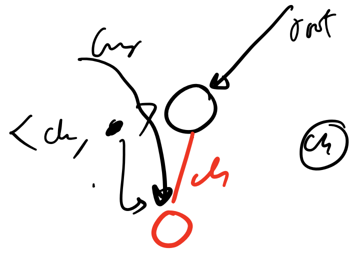
(a) Insert a string in a trie

```
void insert ( TrieNode root, string s) {
    TrieNode cur = root

  for (i=0; i< s.size(); i++) {
      char ch = s[i];
      if (cur.child.find(ch) == false) {
          cur.child[ch] = new TrieNode();
      }
      cur = cur.child[ch];  →  cur.cnt++;

}
```

|s|



```
    cur.isEnd = true;
}
```

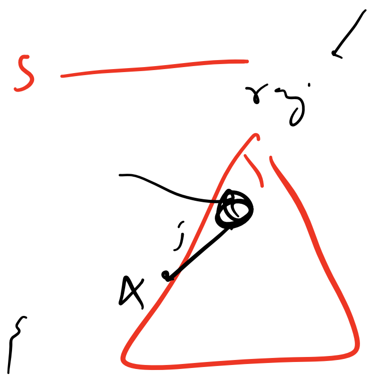
TC = O(|s|)

[illegible] = O(|s|)

required space

④ Check if a string is present or not!

```
bool check(TrieNode root, string s) {
    TrieNode cur = root;
    f(i=0; i< s.size(); i++) {
        char ch = s[i];
        if(cur.child.find(ch)==false) {
            ret false;
        }
        cur = cur.child[ch];
    }
    ret cur.isEnd;
}
```



TC = O(|s|)

SC = O(1)

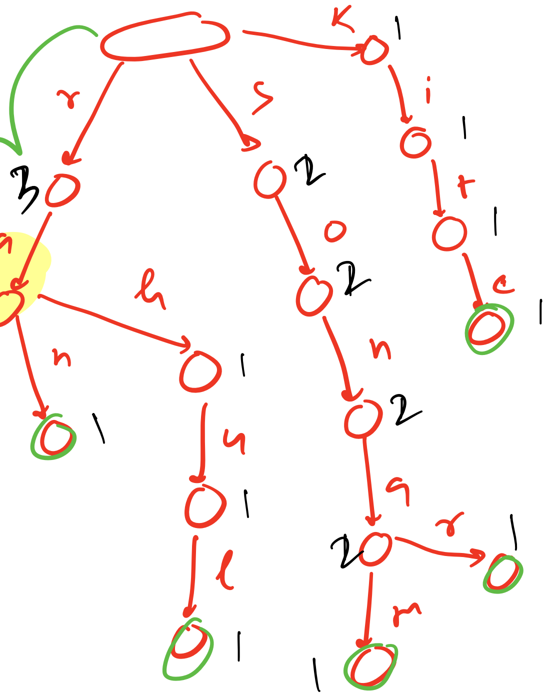
Q Given a set of words & a prefix string. How many words from the set match with the prefix string.

S: [ram, ran, rahul, sonam, sonar, kite]

P: "ra" → 3

"so" → 2

"z" → 0



Trie Node {

~~child;~~

~~isEnd;~~

int cnt;

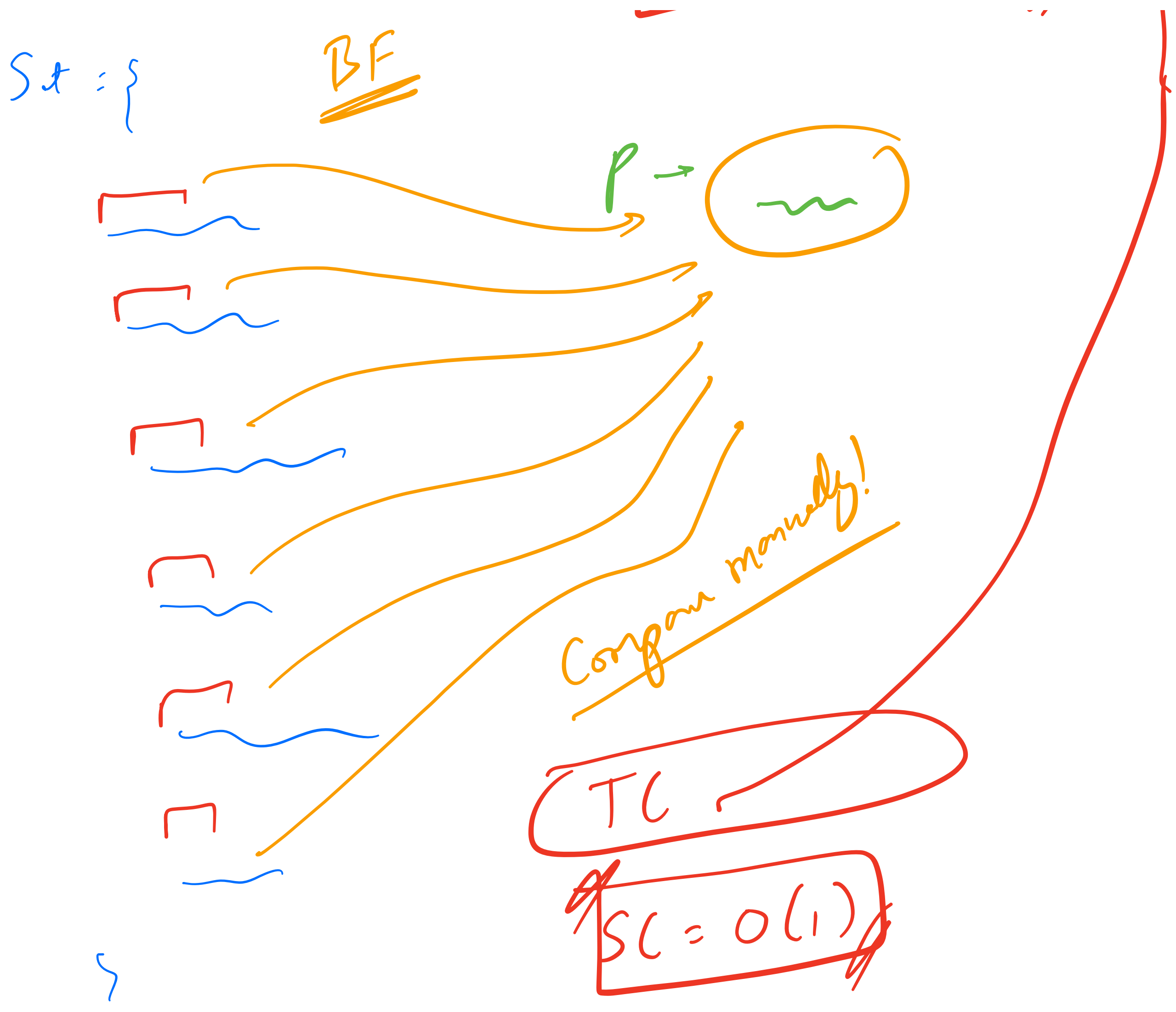
}

total # chars in set → $O(\Sigma |s_i|)$

prefix → $O(|P|)$

$$\boxed{TC = O(\Sigma |s_i| + |P|)}$$

$$\boxed{SC = O(\Sigma |s_i|)}$$



Q SAME QUES AS ABOVE

Q queries : Prefix String.

$$\boxed{TC = O(\Sigma|S_i| + \Sigma|P_i|)}$$

$$\boxed{SC = O(\Sigma|S_i|)}$$

Q SAME QUES AS ABOVE.

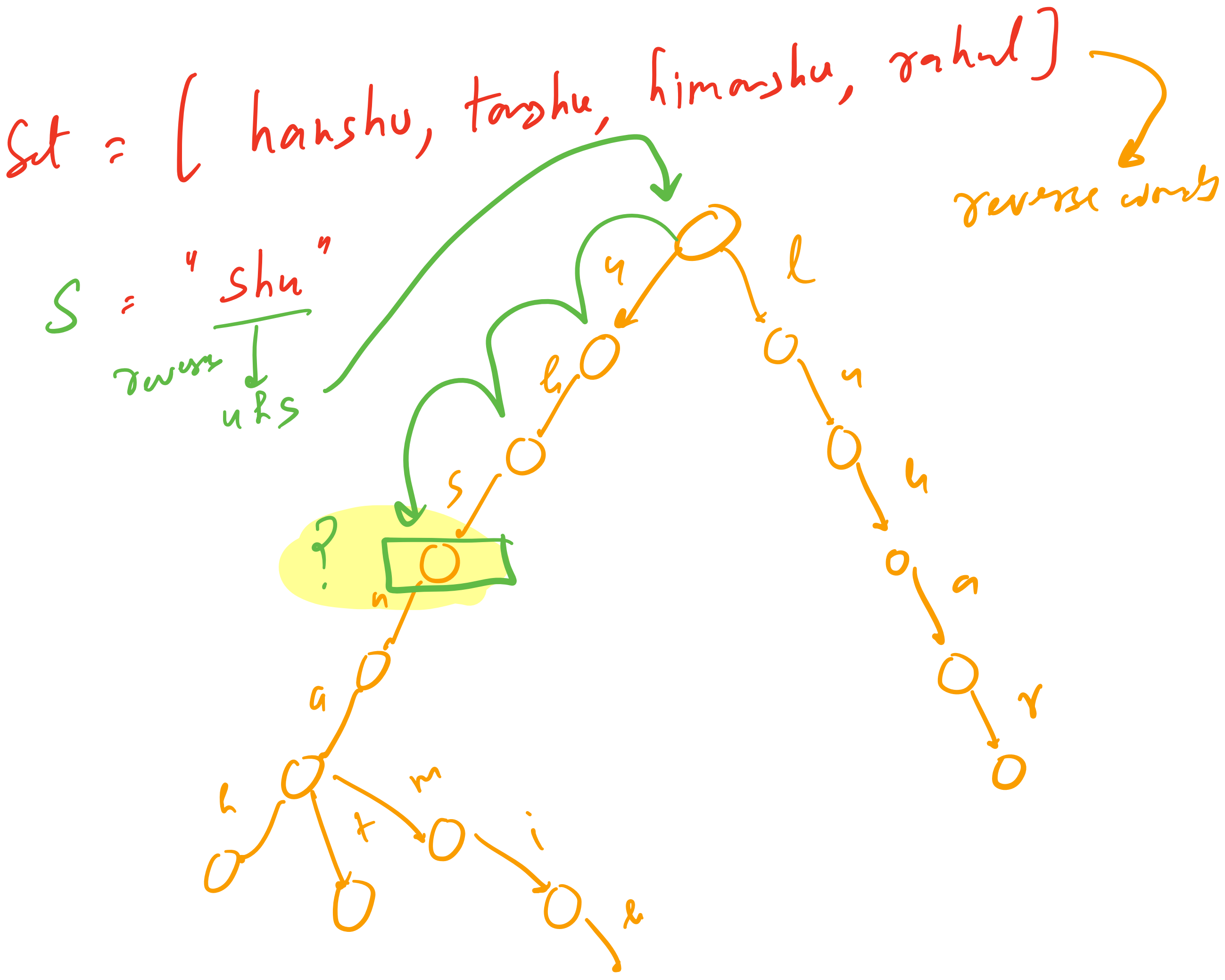
Given a suffix string S.

Check how many words have S as suffix!

Set = [harshu, tanshu, himanshu, rahul]

S = "shu"

reverse ↓ uhs

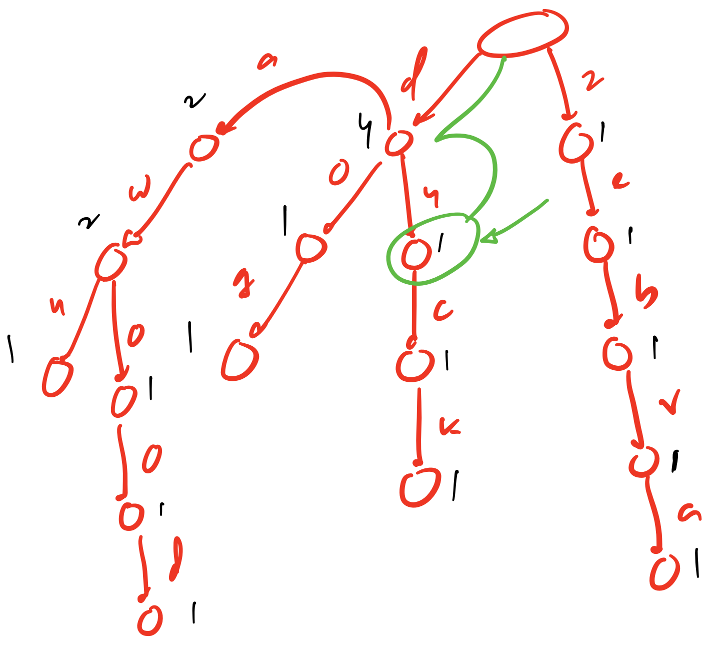


Q Given a set of words.
For each word, tell the shortest prefix that can uniquely identify that word!

NOTE: No word is a prefix of other word!

I/P → S: [dog, zebra, duck, dawn, dawood],

O/P → "do", "z", "du", "dawn", "dawo"]



$\boxed{TC = O(\Sigma|s_i|)}$

$\boxed{SC = O(\Sigma|s_i|)}$