```cpp
#include <bits/stdc++.h>
using namespace std;
#define ll long long
#define inf (int)1e9

```

TrieTree:

```cpp
struct TrieTree{
    struct Node{
        char data; // change type of data
        int wordCount;
        int prefixCount;
        map<char , Node*> edge;

        Node(char a){
            this->data = a;
            this->wordCount = 0;
            this->prefixCount = 0;
        }
    };

    Node* base;

    TrieTree(vector<string> a){
        this->base = new Node('*');
        for (int i = 0; i < a.size(); i++)
            addWord(a[i]);
    }

    void addWord(string s){
        Node* cur = base;
        for (int i = 0; i < s.size(); i++)
        {
            if(cur->edge[s[i]]==NULL) cur->edge[s[i]] = new Node(s[i]);
            cur = cur->edge[s[i]];
            cur->prefixCount++;
        }
        cur->wordCount++;
    }

    // delete word count only not prefix count
    void deleteWord(string s){
        Node* cur = base;
        for (int i = 0; i < s.size(); i++){
            if(cur->edge[s[i]]==NULL) return;
            cur = cur->edge[s[i]];
        }
        if(cur->wordCount!=0)
            cur->wordCount--;
    }

    // return count of word and prefix equal s
    //  word  prefix
    pair<int , int> countWord(string s){
        Node* cur = base;
        for (int i = 0; i < s.size(); i++)
        {
            if(cur->edge[s[i]]==NULL) return {0 , 0};
            cur = cur->edge[s[i]];
        }
        return make_pair(cur->wordCount , cur->prefixCount);
    }
```