

```
1 #include "../bits/stdc++.h"
2 // アルファベット26文字対応のTrie木
3 // 機能拡張, 抽象化したい
4 // verified: https://atcoder.jp/contests/tenka1-2016-final-open/submissions/3501370
5 struct Trie
6 {
7     Trie *node[26];
8     int val;
9     Trie()
10    {
11        val = 0;
12        std::fill(node, node + 26, (Trie *)0);
13    }
14    void insert(const std::string &s, int w)
15    {
16        Trie *r = this;
17        for (int i = 0; i < (int)s.length(); i++)
18        {
19            int c = s[i] - 'a';
20            if (!r->node[c])
21                r->node[c] = new Trie;
22            r = r->node[c];
23        }
24        r->val = w;
25        return;
26    }
27    Trie *find(const char &s, Trie *pos)
28    {
29        int c = s - 'a';
30        if (!pos->node[c])
31            return (Trie *)0;
32        return pos->node[c];
33    }
34 };
35
```