

Trie

维护一个字符串集合，支持两种操作：

1. “I x”向集合中插入一个字符串x；
2. “Q x”询问一个字符串在集合中出现了多少次。

共有N个操作，输入的字符串总长度不超过 10^5 ，字符串仅包含小写英文字母。

输入格式

第一行包含整数N，表示操作数。

接下来N行，每行包含一个操作指令，指令为“I x”或“Q x”中的一种。

输出格式

对于每个询问指令“Q x”，都要输出一个整数作为结果，表示x在集合中出现的次数。

每个结果占一行。

数据范围

$$1 \leq N \leq 2 * 10^4$$

输入样例：

```
5
I abc
Q abc
Q ab
I ab
Q ab
```

输出样例：

```
1
0
1
```

y总代码：

```
#include <iostream>

using namespace std;

const int N = 100010;

int son[N][26], cnt[N], idx;
char str[N];

void insert(char *str)
{
    int p = 0;
    for (int i = 0; str[i]; i ++ )
    {
        int u = str[i] - 'a';
        if (!son[p][u]) son[p][u] = ++ idx;
        p = son[p][u];
    }
    cnt[p] ++ ;
}
```

```

}

int query(char *str)
{
    int p = 0;
    for (int i = 0; str[i]; i++)
    {
        int u = str[i] - 'a';
        if (!son[p][u]) return 0;
        p = son[p][u];
    }
    return cnt[p];
}

int main()
{
    int n;
    scanf("%d", &n);
    while (n--)
    {
        char op[2];
        scanf("%s", op, str);
        if (*op == 'I') insert(str);
        else printf("%d\n", query(str));
    }

    return 0;
}

```

作者: yxc

链接: <https://www.acwing.com/activity/content/code/content/45282/>

来源: AcWing

著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

理解:

```

#include <iostream>
using namespace std;

const int N = 1e5 + 10;
int son[N][26]; // 其中存放的是: 子节点对应的idx。其中son数组的第一维是: 父节点对应的idx,
                // 第二维计数是: 其直接子节点('a' - '0')的值为二维下标。
int cnt[N];     // 以"abc"字符串为例, 最后一个字符---'c'对应的idx作为cnt数组的下标。数组的
                // 值是该idx对应的个数。
int idx;        // 将该字符串分配的一个树结构中, 以下标来记录每一个字符的位置。方便之后的插入
                // 和查找。
char str[N];

void insert(char *str)
{
    int p = 0;
    for (int i = 0; str[i]; i++)
    {
        int u = str[i] - 'a';
        if (!son[p][u]) son[p][u] = ++idx;
        p = son[p][u];
    }
}

```

```

// 此时的p就是str中最后一个字符对应的trie树的位置idx。
cnt[p]++;
}

int query(char *str)
{
    int p = 0;
    for (int i = 0; str[i]; i++)
    {
        int u = str[i] - '0';
        if (!son[p][u]) return 0;
        p = son[p][u];
    }
    return cnt[p];
}

int main()
{
    int n;
    scanf("%d", &n);
    char op[2];
    while (n--)
    {
        scanf("%s%s", op, str);
        if (op[0] == 'I') insert(str);
        else printf("%d\n", query(str));
    }

    return 0;
}

```

作者: goontry

链接: <https://www.acwing.com/solution/content/2083/>

来源: AcWing

著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。