Problem Result

Prefix Suffix Queries

Send Feedback

Ninja has got a string S. He also has two toys: Prefix and Suffix. Toy prefix creates a prefix string out of the given input string and toy suffix creates a suffix string out of the given input string.

For each query string, it can be either prefix or suffix of S, Ninja wants to find out the count of distinct occurrences of the string “ninjas” as a subsequence in the query string. Since Ninja is busy playing with his prefix and suffix toys, therefore, you have to solve this problem.

Input Format:

The first line of input contains a string S. The following line of input contains a number of queries. Let the number of queries be denoted by Q.
Each of the following Q lines contains a character and an integer. The character can either be ‘p’ or ‘s’. ‘p’ denotes that it is a prefix and ‘s’ denotes that it is a suffix. The integer denotes the number of characters of string S involved as prefix or suffix. For example: Let’s suppose that S= “namastefromninjas” and the query is p 8. This denotes that the query string is a prefix of length 8 of the given string S, i.e. “namastef”. Similarly, for query: s 6, the query string will be “ninjas”.

Constraints:

The length of string S will lie in the range: [7, 6000].
All the characters of the string S will be lowercase letters.
The value of Q will lie in the range: [1, 50].
The character in each query will either be ‘p’ or ‘s’.
The integer in each query will lie in the range: [1, length of S].

```
#include<bits/stdc++.h>
using namespace std;
#define ll long long
ll count(string a,string b
{
    ll m=a.size();
    ll n=b.size();
    ll look[m+1][n+1]={{0}
    for(i=0;i<=n;i++)
        look[0][i]=0;
    for(i=0;i<=m;i++)
        look[i][0]=1;
    for(i=1;i<=m;i++)
    {
        for(j=1;j<=n;j++)
        {
            if(a[i-1]==b[j
                look[i][j]
            else
                look[i][j]
        }
    }
    return look[m][n];
}
int main(){
    // write your code
    string s,t;
    cin>>s;
    ll q,i,j,k,n;
    char x;
    cin>>q;
    while(q--)
    {
        cin>>x>>n;
        if(x=='p')
        {
            t=s.substr(0,n
        }
        else
        {
            t=s.substr((ll
        }
        cout<<count(t,"nin
    }
    return 0;
}
//
```