DOUBT SESSION

August 27, 2025

Nephren is playing a game with little leprechauns.

She gives them an infinite array of strings, $f_{0\ldots\infty}$.

f_0 is "What are you doing at the end of the world? Are you busy? Will you save us?".

She wants to let more people know about it, so she defines f_i = "What are you doing while sending "f_{i-1}"? Are you busy? Will you send "f_{i-1}"?" for all $i \geq 1$.

For example, f_1 is

"What are you doing while sending "What are you doing at the end of the world? Are you busy? Will you save us?"? Are you busy? Will you send "What are you doing at the end of the world? Are you busy? Will you save us?"?". Note that the quotes in the very beginning and in the very end are for clarity and are not a part of f_1.

It can be seen that the characters in f_i are letters, question marks, (possibly) quotation marks and spaces.

Nephren will ask the little leprechauns q times. Each time she will let them find the k-th character of f_n. The characters are indexed starting from 1. If f_n consists of less than k characters, output '.' (without quotes).

Can you answer her queries?

$$f_0 = ABC$$
$$f_i = D\ 'f_{i-1}'\ \mathcal{E}\ 'f_{i-1}'$$

$$f_1 = D\ 'f_0'\ \mathcal{E}\ 'f_0'\ = DABC\mathcal{E}ABC$$
$$f_2 = D\ 'f_1'\ \mathcal{E}\ 'f_1'$$
$$= D\ DABC\mathcal{E}ABC\mathcal{E}\ DABC\mathcal{E}ABC$$
$$f_n = D\ f_{n-1}\ \mathcal{E}\ f_{n-1}$$

If we have length of f_{n-1} then we can predict where the K-th character lies.

• **How to find length of $f(n)$ in constant time?**

$$f_0 = 3$$
$$f_1 = 2 + length(f_0)$$
$$f_2 = 2 + length(f_1)$$
$$f_i = 2 + length(f_i)$$

```
int len[100100];
string st = "What are you doing at the end of the world? Are you busy? Wi
string prevv = "What are you doing while sending \"";
string nextt = "\"? Are you busy? Will you send \"";
string lst = "\"?";
void solve()
{
    string f1 = prevv + st + nextt + st + lst;
    cout << f1 << endl;
}

signed main()
{
    ios_base::sync_with_stdio(false);
    cin.tie(0);
// #ifndef ONLINE_JUDGE
//     freopen("input.txt", "r", stdin);
//     freopen("output.txt", "w", stdout);
// #endif // ONLINE_JUDGE
    for (int i = 0; i <= 100000; i++)len[i] = 0;
    len[0] = st.size();
    for (int i = 1; i <= 100000; i++)
    {
        len[i] = prevv.size() + len[i - 1] + nextt.size() + len[i - 1] +
    }
    int t = 1;
    // cin >> t;
    while (t--)
```

Calculated length of fxn

lst.size();

• **Finding Kth character**

5 char   5 char

$$f_i = D\ 'f_{i-1}'\ \mathcal{E}\ 'f_{i-1}'$$

7th character = $\mathcal{E}$
3rd character = 2nd character of f_{i-1}

$$3rd = f_{i-1} = 2nd \quad helper(f_i^o, 3^{rd})$$
$$\hookrightarrow helper(f_{i-1}, 2^{nd})$$

9th character = 2nd character of f_{i-1}
= same helper fxn

```
int len[100100];
string st = "What are you doing at the end of the world? Are you bu
string prevv = "What are you doing while sending \"";
string nextt = "\"? Are you busy? Will you send \"";
string lst = "\"?";
void helper(int n,int k)
{
    //base case
    if(len[n]<k)
    {
        cout<<".";
        return;
    }

    //ith string = st + (i-1)string + nextt + (i-1)thstring + lst

    //lyinh in st part
    if(k<=st.size())
    {
        cout<<st[k-1];
        return;
    }

    k-=st.size();

    //lying in (i-1)th part (first)
    if(k<=len[n-1])
    {
        cout<<st[k-1];
        return;
    }
}
```

```cpp
    k-=len[n-1];

    //lying in next part
    if(k<=nextt.size())
    {
        cout<<nextt[k-1];
        return;
    }
    k-=nextt.size();

    //lying in (i-1)th part (second)
    if(k<=len[n-1])
    {
        helper(n-1,k);
        return;
    }
    k-=len[n-1];

    cout<<"?";
    return;
}

void solve()
{
    int n,k;
    cin>>n>>k;
    helper(n,k);
}
```