CHRL2 - Editorial

3

PROBLEM LINKS

Contest
Practice

2

Author: Roman Furko
Tester: Sergey Kulik
Editorialist: Vinod Reddy

DIFFICULTY:

simple

PREREQUISITES:

basic dp

EXPLANATION

Subtask 1:
We will implement an O(N^2) algorithm for this subtask as N < 2000 this will pass. So the main observation of this algorithm is that consider the 'F' with smallest index such that there exists a subsequence ending at this 'F' with characters 'C','H','E','F'.Now consider such a subsequence T {'C','H','E','F'}. Now this 'F' should be removed from the string in one of the moves in the optimal solution or else if this 'F' is not removed and the 'E' is removed then we can exchange the 'F' removed with the 'E' with our 'F'. if 'E' is also not removed then we can extend our logic to 'H'. if 'H' is not removed then we extend the same logic to 'C'. It cannot be that all of the four characters are not removed.

Now when we remove this 'F' in a move it can always be removed along with the nearest 'E' to the left of it because if it's not the case either 'E' is not removed at all in which case the presently used 'E' can be replaced with the nearest 'E' or this 'E' is used with another 'F' in which case the 'CHE' of second 'F' can be exchanged with 'CHE' of the first. Similarly we 'H' be the nearest one to the left of chosen 'E' and similar case for 'C'. So we find such subsequence and remove those four characters and repeat the same procedure for remaining string
So for example lets take the string CHECHFECF.

```
1 2 3 4 5 6 7 8 9
C H E C H F E C F
```

The first 'F' with a sequence possible will be the one at position 6. And the corresponding 'E' will be the one at position 3 and 'H' at 2 and C at 1. When we remove the characters we will be left with CHECF and again we follow same logic and remove one more 'CHEF'.

Implementation : It can be implemented in O(N^2). You can use a boolean array to track whether a symbol is removed or not and for each 'F' from left to right try to find the first 'E' to the left of it which is not removed and then first 'H' which not removed to the left of 'E' and then 'C'. Then remove these characters by making their tracking variables to true. It's easy to implement. For the code please check editorials solution.

Subtask 2 : This is almost same as above algorithm but just a good implementation.

Since at every time in the above algorithm we check for the nearest next character to left which is not used (for 'F' we check for nearest left 'E',for 'E' we check for 'H',for 'H' we check for 'C') we will precompute the information. We will maintain four vectors one each for positions of each of characters 'C','H','E','F'. so the arrays will be P[char][int[]].

We loop from left to right. Whenever 'C' comes we insert it's position into P['C'] . Whenever we encounter 'H' we check if P['C'] is empty. If it's empty then this 'H' is not useful as we cannot use it to form a subsequence so we throw it away. If P['C'] is not empty then we can map 'H' to the last position in P['C'] and so whenever 'H' is used we use it with 'C'. Since the 'C' is mapped to 'H' we just remove the last element of P['C']. So the array P['H'] are positions of 'H' for which there is a 'C' mapping and not mapped to any 'E'. whenever 'E,'F'' comes we do the same thing as 'H'. So finally the elements of C['F'] contains positions of 'F' for which there is a mapping to 'E' and this 'E' has a mapping to 'H' and this 'H' has a mapping to 'C'.So finally no of elements in C['F'] will be our answer.

Here a simple observation will help. We really need not maintain the positions in P[char][int[]]. We just need to know the size of P['C'],P['H'],P['E'],P['F'].So now new array will be NP[char][int] .When removing element originally in P['C'],P['E'],P['H'] we just decrease the value in NP['C'],NP['H'],NP['E']. When we were inserting before we just increase the value against character. See the setter's code for exact implementation.

SOLUTIONS

Setter's Solution: CHRL2.cpp
Tester's Solution: CHRL2.cpp
Editorialist's Solution: CHRL2.cpp

ltime08 chrl3 easy dp

This question is marked "community wiki".

edited **26 Jan '14, 14:20**

asked **26 Jan '14, 14:19**
vaka ♦♦
243●13●18●22
accept rate: 16%

Tags:

easy **×1,220**
dp **×675**
ltime08 **×8**
chrl3 **×5**

Asked: **26 Jan '14, 14:19**
Seen: **2,180 times**
Last updated: **04 Sep, 02:30**

Related questions

- CHRL3 - Editorial
- DP easy problems
- SANSKAR - Editorial
- Help with basic dp!!
- SEARRAYS - Editorial
- DIAMOND - Editorial
- CHEFBR - Editorial
- PPTEST - Editorial
- Help needed in AEHASH
- MAXPR - Editorial

6 Answers: oldest newest **most voted**

11

```
#include< iostream>

#include< cstdio>

#include< cstring>

char a[100005];

int main(){

   scanf("%s",a);

   int i,c,ch,che,chef;
   c=ch=che=chef=0;

   for(i=0;i<strlen(a);i++){
       if(a[i]=='C'){
           c++;
       }
       else if(a[i]=='H'){
           if(c>0){c--; ch++;}
       }
       else if(a[i]=='E'){
           if(ch>0){ch--; che++;}
       }
       else if(a[i]=='F'){
           if(che>0){che--; chef++;}
       }
   }
   printf("%d\n",chef);
   return 0;
}
```

link | award points

edited **27 Jan '14, 04:34**

betlista ♦♦
16.6k●49●114●222

answered **26 Jan '14, 15:23**

ggeek
91●1●1●2
accept rate: 0%

oh I had it like you but without `c--` in the `if str[i] == 'H'`, I had instead `if str[i] == 'F' { c--; h--; e--; f++}` which was wrong :P

goodwine (26 Jan '14, 23:27)

very neat code ;-)

betlista ♦♦ (27 Jan '14, 04:35)

glad u liked it.. :D

ggeek (27 Jan '14, 18:01)

had exactly the same code (though the name of variables are different) :-) very good writing style .. easy to understand..

deepu_codes (27 Jan '14, 21:30)

0

Tell me if my approach is wrong.Aren't we just calculating the number of time the word CHEF can be formed using those characters only once?..if yes then my logic is to just count the number of c's ,f's,e's and h's and the minimum of them is the count we require.

THANK YOU

```
c,h,e,f=0,0,0,0
A=raw_input()
for i in range(0,len(A)):
    if A[i]=="C":
        c+=1
    elif A[i]=="E":
        e+=1
    elif A[i]=="F":
        f+=1
    elif A[i]=="H":
        h+=1
x=min(c,e,f,h)
print(x)

edit:
Do we require the CHEF to be in the same order in the given string?...
```

link | award points

edited **05 Oct '14, 16:15**

answered **05 Oct '14, 16:06**

broadsword
1●1
accept rate: 0%

consider , FEHCCHEF , ANS=1 but ur ANS=2

empty_life (31 Aug, 06:32)

@broadword: did u get the answer to your query...I am also having the same query. i also used the same algo.

0

link | award points

answered **24 Feb, 17:38**

ddevil
1
accept rate: 0%

include<iostream>

0 **include<string.h>**

using namespace std; int main() { char a[100000]; char b[20]="CHEF"; char *p=b; int c=0,i; int d[100000]; cin>>a; for(i=0;i<100000;i++) d[i]=0; for(i=0;i<strlen(a);i++)

if(d[i]==0) if(a[i]==*p) if*(p=='F') {p=b; c++; i=0; d[i]=1; } else { p++; d[i]=1;}

cout<<c;

return 0;
}

link | award points

answered **04 Sep, 02:30**

rohitranjan017
1
accept rate: 0%

Noida Packers and movers @ http://www.noidapackers.co.in/

-1

link | award points

answered **01 Sep, 10:41**

b2bad
-17
accept rate: 0%

Delhi Packers and Movers Noida Packers and Movers

-1 http://b2bad.in/Packers-and-Movers-in-Gurgaon/">Gurgaon packers and Movers http://b2bad.in/Packers-and-Movers-in-Faridabad/">Faridabad packers and Movers http://b2bad.in/Packers-and-Movers-in-Indirapuram-Ghaziabad/ ">Ghaziabad packers and Movers

link | award points

answered **01 Sep, 10:41**

b2bad
-17
accept rate: 0%

Your answer

[hide preview] community wiki

and

Type the text

Privacy & Terms

Post Your Answer