Strings ☆



X

Your Strings submission got 10.00 points.

Share

Tweet

You are now 50 points away from the 4th star for your c++ badge.

Try the next challenge | Try a Random Challenge

Problem

Submissions

Leaderboard

C++ provides a nice alternative data type to manipulate strings, and the data type is conveniently called string. Some of its widely used features are the following:

• Declaration:

```
string a = "abc";
```

• Size:

```
int len = a.size();
```

• Concatenate two strings:

```
string a = "abc";
string b = "def";
string c = a + b; // c = "abcdef".
```

Accessing ith element:

```
string s = "abc";
char c0 = s[0];  // c0 = 'a'
char c1 = s[1];  // c1 = 'b'
char c2 = s[2];  // c2 = 'c'
s[0] = 'z';  // s = "zbc"
```

P.S.: We will use cin/cout to read/write a string.

Input Format

You are given two strings, $m{a}$ and $m{b}$, separated by a new line. Each string will consist of lower case Latin characters ('a'-'z').

Output Format

In the first line print two space-separated integers, representing the length of \boldsymbol{a} and \boldsymbol{b} respectively.

In the second line print the string produced by concatenating a and b (a + b).

In the third line print two strings separated by a space, a' and b'. a' and b' are the same as a and b, respectively, except that their first characters swapped.

Sample Input

```
abcd
ef

Sample Output

4 2
abcdef
ebcd af

Explanation

• a = "abcd"
• b = "ef"

• |a| = 4
• |b| = 2
• a + b = "abcdef"
• a' = "ebcd"
• b' = "af"
```

```
C++
                                                             Change Theme
                                                                                                      10 [2] ($\dip$
     #include <iostream>
1
    #include <cstring>
2
3
    #include <string>
 4
     #include <string.h>
 5
    using namespace std;
6
7
     int main() {
8
         // Complete the program
9
         string str1,str2;//here we have taken a predefined class string
10
         cin>>str1>>str2;
11
         cout<<str1.size()<<" "<<str2.size()<<endl;//size() works with string class.If we</pre>
     take char array to store string then size() function will not work.
12
         cout<<str1+str2<<endl;</pre>
         \verb|cout|<<|str2.substr(0,1)|+|str1.substr(1,str1.size()-1)|<<||"|<<|str1.substr(0,1)|
13
     +str2.substr(1,str2.size()-1)<<endl;</pre>
14
15
         return 0;
     }
16
                                                                                                    Line: 14 Col: 3
```

 $oldsymbol{\uparrow}$ Upload Code as File $oxedsymbol{\Box}$ Test against custom input

Run Code

Submit Code

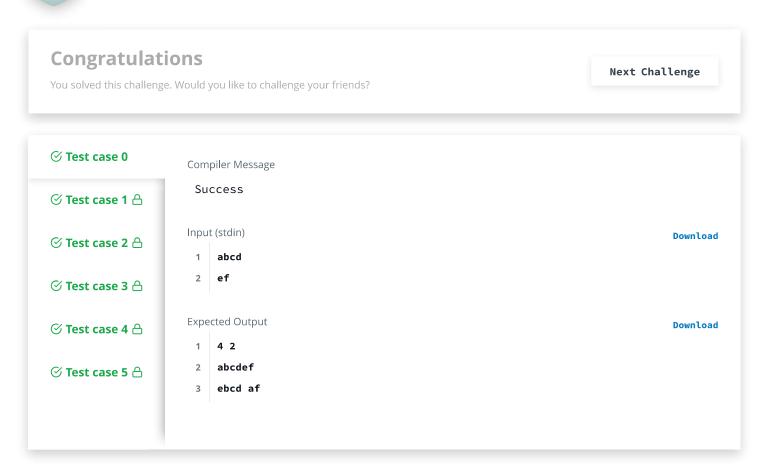
You have earned 10.00 points!

You are now 50 points away from the 4th star for your c++ badge.









Contest Calendar | Blog | Scoring | Environment | FAQ | About Us | Support | Careers | Terms Of Service | Privacy Policy | Request a Feature

