## 将字符串str中的str1替换为str2.

|  |
| --- |
| #include <iostream>  #include <vector>  #include <string>  #include <algorithm>  **using** **namespace** std**;**  string strReplace**(**string**&** str**,**const string**&** str1**,**const string**&** str2**){**  **if(**str**.**empty**()){**  **return** str**;**  **}**  int len **=** str**.**size**();**  int len1 **=** str1**.**size**();**  int len2 **=** str2**.**size**();**  int diff **=** len1**-**len2**;**  diff **=** diff**>=**0**?**0**:-**diff**;**  string tmp **=** str**;**  transform**(**tmp**.**begin**(),**tmp**.**end**(),**tmp**.**begin**(),::**tolower**);**  vector**<**int**>** pos**;**  int i **=** **-**1 **;**  **while((**i **=** tmp**.**find**(**str1**,**i**+**1**))** **!=** **-**1**){**  pos**.**push\_back**(**i**+**diff**\***pos**.**size**());**  **}**  **if(**diff**==**0**){**  **for(**auto**&** a**:**pos**){**  **for(** int i **=** 0**;** i **<** len1 **;++**i**){**  str**[**a**+**i**]** **=** i **<** len2**?**str2**[**i**]:**'#'**;**  **}**  **}**  **for(**int i **=** 0**;** i **<** len **;++**i**){**  **if(**str**[**i**]** **==** '#'**){**  str**.**erase**(**str**.**begin**()+**i**);**  **--**i**;**  **}**  **}**  **}**  **else{**  **for(**auto**&** a**:**pos**){**  int i **=** 0**;**  **for(;** i **<** len1**;** **++**i**){**  str**[**a**+**i**]** **=** str2**[**i**];**  **}**  **for(;**i**<**len2 **;++**i**){**  str**.**insert**(**str**.**begin**()+**a**+**i**,**str2**[**i**]);**  **}**  **}**  **}**  **return** str**;**  **}**  int main**()**  **{**  vector**<**string**>** vec**;**  string tmp**;**  **while(**getline**(**cin**,**tmp**)){**  // if(tmp == "#"){  // break;  // }  vec**.**push\_back**(**strReplace**(**tmp**,**"marshtomp"**,**"fjxmlhx"**));**  tmp **=** ""**;**  **}**  **for(**auto**&** a**:** vec**){**  cout**<<**a**<<**endl**;**  **}**  **return** 0**;**  **}** |

## 并查集

|  |
| --- |
| #include <iostream>  #include <vector>  #include <string>  #include <algorithm>  #include <map>  #include <set>  #include <stdio.h>  **using** **namespace** std**;**  string mfind**(**map**<**string**,**pair**<**string**,**int**>>&** mmap**,**string name**){**  **if(**mmap**.**empty**()||**mmap**.**find**(**name**)** **==** mmap**.**end**()){**  **return** ""**;**  **}**  **if(**mmap**[**name**].**first **==** name**){**  **return** name**;**  **}**  **return** mfind**(**mmap**,**mmap**[**name**].**first**);**  **}**  void minsert**(**map**<**string**,**pair**<**string**,**int**>>&** mmap**,**string name1**,**string name2**){**  **if(**mmap**.**find**(**name1**)** **==** mmap**.**end**()){**  mmap**[**name1**].**first **=** name1**;**  mmap**[**name1**].**second **=** 1**;**  **}**  **if(** mmap**.**find**(**name2**)** **==** mmap**.**end**()){**  mmap**[**name2**].**first **=** name2**;**  mmap**[**name2**].**second **=** 1**;**  **}**  string f1 **=** mfind**(**mmap**,**name1**);**  string f2 **=** mfind**(**mmap**,**name2**);**  **if(**f1 **!=** f2 **){**  **if(**mmap**[**f1**].**second**>=**mmap**[**f2**].**second**){**  mmap**[**f2**].**first **=** f1**;**  mmap**[**f1**].**second**++;**  **}**  **else{**  mmap**[**f1**].**first **=** f2**;**  mmap**[**f2**].**second**++;**  **}**  **}**  **}**  int main**()**  **{**  int n **=** 0**;**  cin**>>**n**;**  vector**<**bool**>** ret**;**  map**<**string**,**pair**<**string**,**int**>>** mmap**;**  **for(**int i **=** 0 **;** i **<** n**;++**i**){**  int op **=** 0**;**  string name1**;**  string name2**;**  cin**>>**op**>>**name1**>>**name2**;**  **if(**op **==** 0**){**  minsert**(**mmap**,**name1**,**name2**);**  **}**  **else{**  string f1 **=** mfind**(**mmap**,**name1**);**  string f2 **=** mfind**(**mmap**,**name2**);**  ret**.**push\_back**(!**f1**.**empty**()** **&&** **!**f2**.**empty**()** **&&** f1 **==** f2**);**  **}**  **}**  **for(**auto**&** a**:**mmap**){**  cout**<<**a**.**first**<<**" "**<<**a**.**second**.**first**<<**" "**<<**a**.**second**.**second**<<**endl**;**  **}**  **for(**auto a**:** ret**){**  **if(**a**){**  cout**<<**"yes"**<<**endl**;**  **}**  **else{**  cout**<<**"no"**<<**endl**;**  **}**  **}**  **return** 0**;**  **}** |

## 扩展的二进制：

|  |
| --- |
| #include <iostream>  #include <vector>  #include <string>  #include <algorithm>  #include <map>  #include <set>  #include <stdio.h>  #include <bitset>  **using** **namespace** std**;**  int helper**(**int n**){**  **if(**n **==** 0**){**  **return** 1**;**  **}**  **if(**n**&**0x1**){**  **return** helper**(**n**>>**1**);**  **}**  **return** helper**(**n**>>**1**)+**helper**((**n**>>**1**)-**1**);**  **}**  int main**()**  **{**  int n **=** 0**;**  cin**>>**n**;**  cout**<<**helper**(**n**);**  **return** 0**;**  **}** |

## 简单计算器

|  |
| --- |
| #include <iostream>  #include <stack>  **using** **namespace** std**;**  bool mf**(**stack**<**int**>&** num**,**stack**<**char**>&** op**,**int level**){**  **switch(**level**){**  **case** 1**:**  **while(!**op**.**empty**()** **&&** **(**op**.**top**()** **==** '\*' **||** op**.**top**()** **==** '/'**)){**  **if(**num**.**size**()** **<** 2 **){**  **return** **false;**  **}**  int num1 **=** num**.**top**();**  num**.**pop**();**  int num2 **=** num**.**top**();**  num**.**pop**();**  int sum **=** 0**;**  char oper **=** op**.**top**();**  op**.**pop**();**  **switch(**oper**){**  **case** '\*'**:**  sum **=** num1 **\*** num2**;**  **break;**  **case** '/'**:**  sum **=** num2 **/** num1**;**  **break;**  **default:break;**  **}**  num**.**push**(**sum**);**  **}**  **break;**  **case** 2**:**  **while(!**op**.**empty**()** **&&** op**.**top**()** **!=** '('**){**  **if(**num**.**size**()** **<** 2 **){**  **return** **false;**  **}**  int num1 **=** num**.**top**();**  num**.**pop**();**  int num2 **=** num**.**top**();**  num**.**pop**();**  int sum **=** 0**;**  char oper **=** op**.**top**();**  op**.**pop**();**  **switch(**oper**){**  **case** '\*'**:**  sum **=** num1 **\*** num2**;**  **break;**  **case** '/'**:**  sum **=** num2 **/** num1**;**  **break;**  **case** '+'**:**  sum **=** num2 **+** num1**;**  **break;**  **case** '-'**:**  sum **=** num2 **-** num1**;**  **break;**  **default:break;**  **}**  num**.**push**(**sum**);**  **}**  **break;**  **default:break;**  **}**  **return** **true;**  **}**  int helper**(**string str**){**  **if(**str**.**empty**()){**  **return** 0**;**  **}**  str **+=** "#"**;**  stack**<**int**>** num**;**  stack**<**char**>** op**;**  int len **=** str**.**size**();**  int i **=** 0**;**  **while(**i **<** len**){**  **if(**str**[**i**]** **>=** '0' **&&** str**[**i**]** **<=** '9'**){**  int tmp **=** 0**;**  **for(;**i**<**len**;++**i**){**  **if(**str**[**i**]** **>=** '0' **&&** str**[**i**]** **<=** '9'**){**  tmp **=** tmp**\***10 **+** str**[**i**]-**'0'**;**  **}**  **else{**  **break;**  **}**  **}**  num**.**push**(**tmp**);**  **}**  **switch(**str**[**i**]){**  **case** '('**:**  op**.**push**(**'('**);**  **break;**  **case** ')'**:**  mf**(**num**,**op**,**2**);**  op**.**pop**();**  **break;**  **case** '\*'**:**  **case** '/'**:**  mf**(**num**,**op**,**1**);**  op**.**push**(**str**[**i**]);**  **break;**  **case** '+'**:**  **case** '-'**:**  **case** '#'**:**  mf**(**num**,**op**,**2**);**  op**.**push**(**str**[**i**]);**  **break;**  **}**  **++**i**;**  **}**  **if(**num**.**size**()** **==** 1 **&&** op**.**size**()** **==** 1**){**  **return** num**.**top**();**  **}**  **return** 0**;**  **}**  int main**()**  **{**  string s**;**  cin**>>**s**;**  cout**<<**helper**(**s**);**  **return** 0**;**  **}** |