831. Masking Personal Information

DescriptionHintsSubmissionsDiscussSolution

We are given a personal information string `S`, which may represent either **an email address** or **a phone number.**

We would like to mask this personal information according to the following rules:

1. Email address:

We define a **name** to be a string of `length ≥ 2` consisting of only lowercase letters `a-z` or uppercase letters `A-Z`.

An email address starts with a name, followed by the symbol `'@'`, followed by a name, followed by the dot `'.'` and followed by a name.

All email addresses are guaranteed to be valid and in the format of `"name1@name2.name3".`

To mask an email, **all names must be converted to lowercase** and **all letters between the first and last letter of the first name** must be replaced by 5 asterisks `'*'`.

2. Phone number:

A phone number is a string consisting of only the digits `0-9` or the characters from the set `{'+', '-', '(', ')', ' '}.` You may assume a phone number contains 10 to 13 digits.

The last 10 digits make up the local number, while the digits before those make up the country code. Note that the country code is optional. We want to expose only the last 4 digits and mask all other digits.

The local number should be formatted and masked as `"***-***-1111",` where `1` represents the exposed digits.

To mask a phone number with country code like `"+111 111 111 1111"`, we write it in the form `"+***-***-***-1111".` The `'+'` sign and the first `'-'` sign before the local number should only exist if there is a country code. For example, a 12 digit phone number mask should start with `"+**-"`.

Note that extraneous characters like `"(", ")", " "`, as well as extra dashes or plus signs not part of the above formatting scheme should be removed.

Return the correct "mask" of the information provided.

Example 1:

```
Input: "LeetCode@LeetCode.com"
Output: "l*****e@leetcode.com"
Explanation: All names are converted to lowercase, and the letters between the
             first and last letter of the first name is replaced by 5 asterisks.
             Therefore, "leetcode" -> "l*****e".
```

Example 2:

```
Input: "AB@qq.com"
Output: "a*****b@qq.com"
Explanation: There must be 5 asterisks between the first and last letter
             of the first name "ab". Therefore, "ab" -> "a*****b".
```

Example 3:

```
Input: "1(234)567-890"
Output: "***-***-7890"
Explanation: 10 digits in the phone number, which means all digits make up the
local number.
```

Example 4:

```
Input: "86-(10)12345678"
Output: "+**-***-***-5678"
Explanation: 12 digits, 2 digits for country code and 10 digits for local
number.
```

Notes:

1. `S.length <= 40`.
2. Emails have length at least 8.
3. Phone numbers have length at least 10.

Seen this question in a real interview before?

- Subscribe to see which companies asked this question.

  Related Topics String

A StraightForward Solution

```
class Solution {
public:

string phone(string &S){
   string ans="",digits="";
   for(int i=0;i<(int)S.size();++i)
   {
      if(S[i]-'0'>=0 && S[i]-'0'<=9)
      {
         digits+=S[i];
      }
   }
   if(digits.size()>10)
   {
      ans+="+"+string(digits.size()-10,'*')+"-";
   }
   ans += "***-***-" + digits.substr(digits.size()-4);
   return ans;
}


string email(string &S, int pos)
{
   string ans=S.substr(0,1)+"*****"+S.substr(pos-1);
   string ret = "";
   for(char c:ans)
   {
      char d = tolower(c);
      ret+=d;
   }
   return ret;
}

string maskPII(string S) {
       int pos=S.find('@');
       if (pos==(int)string::npos)
           return phone(S);
       else
           return email(S,(int)pos);
    }

};
```