Articles > 929. Unique Email Addresses

Previous (/articles/minimum-path-falling-sum/) Next (/articles/binary-subarrays-with-sum/)

929. Unique Email Addresses (/problems/unique-email-addresses/)

Oct. 27, 2018 | 112.2K views

Average Rating: 2.50 (109 votes)

Every email consists of a local name and a domain name, separated by the @ sign.

For example, in `alice@leetcode.com` , `alice` is the local name, and `leetcode.com` is the domain name.

Besides lowercase letters, these emails may contain `'.'` s or `'+'` s.

If you add periods (`'.'`) between some characters in the **local name** part of an email address, mail sent there will be forwarded to the same address without dots in the local name. For example, `"alice.z@leetcode.com"` and `"alicez@leetcode.com"` forward to the same email address. (Note that this rule does not apply for domain names.)

If you add a plus (`'+'`) in the **local name**, everything after the first plus sign will be **ignored**. This allows certain emails to be filtered, for example `m.y+name@email.com` will be forwarded to `my@email.com` . (Again, this rule does not apply for domain names.)

It is possible to use both of these rules at the same time.

Given a list of `emails` , we send one email to each address in the list. How many different addresses actually receive mails?

Example 1:

```
Input: ["test.email+alex@leetcode.com","test.e.mail+bob.cathy@leetcode.com
Output: 2
Explanation: "testemail@leetcode.com" and "testemail@lee.tcode.com" actual
```

Note:

- `1 <= emails[i].length <= 100`
- `1 <= emails.length <= 100`
- Each `emails[i]` contains exactly one `'@'` character.
- All local and domain names are non-empty.
- Local names do not start with a `'+'` character.

Solution

Approach 1: Canonical Form

Intuition and Algorithm

For each email address, convert it to the *canonical* address that actually receives the mail. This involves a few steps:

- Separate the email address into a `local` part and the `rest` of the address.

- If the `local` part has a `'+'` character, remove it and everything beyond it from the `local` part.

- Remove all the zeros from the `local` part.

- The canonical address is `local + rest` .

After, we can count the number of unique canonical addresses with a `Set` structure.

Java Python Copy

```java
class Solution {
  public int numUniqueEmails(String[] emails) {
    Set<String> seen = new HashSet();
    for (String email : emails) {
      int i = email.indexOf('@');
      String local = email.substring(0, i);
      String rest = email.substring(i);
      if (local.contains("+")) {
        local = local.substring(0, local.indexOf('+'));
      }
      // Note: one should escape the specific character '.',
      // since it is treated as a regex expression.
      local = local.replaceAll("\\.", "");
      seen.add(local + rest);
    }

    return seen.size();
  }
}
```

Complexity Analysis

- Time Complexity: $O(\mathcal{C})$, where $\mathcal{C}$ is the total content of `emails` .
- Space Complexity: $O(\mathcal{C})$.

Rate this article:

Previous (/articles/minimum-path-falling-sum/)

Next (/articles/binary-subarrays-with-sum/)

Comments: 75

Sort By

Type comment here... (Markdown is supported)

Preview Post

shuaizki (/shuaizki) ★ 95 March 31, 2019 7:07 AM

```
        local = local.replaceAll(".", "");
```

is wrong here, should use

Read More

74 ⌃ ⌄ | Share | Reply

SHOW 4 REPLIES

bobxu1128 (/bobxu1128) ★ 187 December 28, 2018 1:16 AM

time complexity is questionable.

28 ⌃ ⌄ | Share | Reply

SHOW 3 REPLIES

mrdavidlong (/mrdavidlong) ★ 44 March 4, 2019 6:40 AM

On line 11, the dot character needs to be escaped. It should be :

```
local = local.replaceAll("\\.", "");
```

Read More

24 ⌃ ⌄ | Share | Reply

SHOW 6 REPLIES

evil_zone321 (/evil_zone321) ★ 89 February 9, 2019 4:34 AM

Above code fails to give correct result for this test case:

["testemail@leetcode.com","testemail1@leetcode.com","testemail+david@lee.tcode.com"]

24 ⌃ ⌄ | Share | Reply

SHOW 2 REPLIES

TimothyLiu (/timothyliu) ★ 26 March 31, 2019 10:40 AM

This solution is misleading, since too many library methods are called

16 ⌃ ⌄ | Share | Reply

Ruotianzhang (/ruotianzhang) ★ 14 November 3, 2018 4:03 AM

In fact, it should use String.replace(), instead of String.replaceAll(). Cause replaceAll() treats "." like a regex so that it would simple replace every character to "", and the result would only contain domain String.

14 ⌃ ⌄ | Share | Reply

SHOW 1 REPLY

khamid (/khamid) ★ 18 February 12, 2019 3:30 AM

Time complexity is definetly not O(C), String contains method and replaceAll are going through the array of chars in String. So it will be O(C * J) where J is the number of chars in String.

11 ⌃ ⌄ | Share | Reply

SHOW 5 REPLIES

krishreddy56 (/krishreddy56) ★ 6 February 16, 2019 12:12 AM

Use replace instead of replaceAll for replacing . with empty string

6 ⌃ ⌄ Share Reply

SHOW 2 REPLIES

RobynZhao (/robynzhao) ★ 6 February 13, 2019 9:05 AM

we need add a if condition for line 11, such as
if(local.contains(".")) {
local = local.replaceAll(".", "");
}

6 ⌃ ⌄ Share Reply

SHOW 1 REPLY

zhangleirunning (/zhangleirunning) ★ 131 December 31, 2018 1:45 PM

local = local.replaceAll(".", ""); // WRONG! local will be empty

6 ⌃ ⌄ Share Reply

SHOW 4 REPLIES

‹ 1 2 3 4 5 6 7 8 ›