Isomorphic Strings

Given two strings ***s*** and ***t***, determine if they are isomorphic.

Two strings are isomorphic if the characters in ***s*** can be replaced to get ***t***.

All occurrences of a character must be replaced with another character while preserving the order of characters. No two characters may map to the same character but a character may map to itself.

For example,
Given `"egg"`, `"add"`, return true.

Given `"foo"`, `"bar"`, return false.

Given `"paper"`, `"title"`, return true.

Note:
You may assume both ***s*** and ***t*** have the same length.

Solution 1

```
class Solution {
public:
    bool isIsomorphic(string s, string t) {
        int m1[256] = {0}, m2[256] = {0}, n = s.size();
        for (int i = 0; i < n; ++i) {
            if (m1[s[i]] != m2[t[i]]) return false;
            m1[s[i]] = i + 1;
            m2[t[i]] = i + 1;
        }
        return true;
    }
};
```

written by grandyang original link here

Solution 2

Hi guys!

The main idea is to store the last seen positions of current (i-th) characters in both strings. If previously stored positions are different then we know that the fact they're occuring in the current i-th position simultaneously is a mistake. We could use a map for storing but as we deal with chars which are basically ints and can be used as indices we can do the whole thing with an array.

Check the code below. Happy coding!

```
public class Solution {
    public boolean isIsomorphic(String s1, String s2) {
        int[] m = new int[512];
        for (int i = 0; i < s1.length(); i++) {
            if (m[s1.charAt(i)] != m[s2.charAt(i)+256]) return false;
            m[s1.charAt(i)] = m[s2.charAt(i)+256] = i+1;
        }
        return true;
    }
}
```

written by shpolsky original link here

Solution 3

```
bool isIsomorphic(char* s, char* t) {
    char charArrS[256] = { 0 };
    char charArrT[256] = { 0 };
    int i = 0;
    while (s[i] !=0)
    {
        if (charArrS[s[i]] == 0 && charArrT[t[i]] == 0)
        {
            charArrS[s[i]] = t[i];
            charArrT[t[i]] = s[i];
        }
        else
        if (charArrS[s[i]] != t[i] || charArrT[t[i]] != s[i])
            return false;
        i++;
    }

    return true;
}
```

written by rxm24217 original link here