

205-同构字符串

题述

205. 同构字符串

难度 简单  520     

给定两个字符串 `s` 和 `t`，判断它们是否是同构的。

如果 `s` 中的字符可以按某种映射关系替换得到 `t`，那么这两个字符串是同构的。

每个出现的字符都应当映射到另一个字符，同时不改变字符的顺序。不同字符不能映射到同一个字符上，相同字符只能映射到同一个字符上，字符可以映射到自己本身。

示例 1:

输入: `s = "egg", t = "add"`
输出: `true`

示例 2:

输入: `s = "foo", t = "bar"`
输出: `false`

示例 3:

输入: `s = "paper", t = "title"`
输出: `true`

思路

- 题述信息中给出，字符串没有说都是小写等类似说明，使用map作哈希映射
- 使用两个map保存s[i]到t[i]和t[j]到s[i]的映射关系，如果发现对应不上，立刻返回false

题解

哈希

```
class Solution {
public:
    bool isIsomorphic(string s, string t)
    {
        unordered_map<char, char> map1;
        unordered_map<char, char> map2;
        for(int i=0, j=0; i<s.size(); i++, j++)
        {
```

```

        if(map1.find(s[i])==map1.end())
        {
            // map1保存s[i] 到 t[j]的映射
            map1[s[i]] = t[j];
        }
        if(map2.find(t[j])==map2.end())
        {
            // map2保存t[j] 到 s[i]的映射
            map2[t[j]] = s[i];
        }
        //发现映射 对应不上情况，则立即返回false
        if(map1[s[i]]!=t[j] || map2[t[j]]!=s[i])
        {
            return false;
        }
    }
    return true;
}
};

```

思考

执行结果: **通过** [显示详情](#) >

[添加备注](#)

执行用时: **8 ms** , 在所有 C++ 提交中击败了 **59.31%** 的用户

内存消耗: **6.8 MB** , 在所有 C++ 提交中击败了 **74.84%** 的用户

通过测试用例: **43 / 43**

炫耀一下:



[写题解，分享我的解题思路](#)

提交结果	执行用时	内存消耗	语言	提交时间	备注
通过	8 ms	6.8 MB	C++	2022/09/21 13:22	添加备注