1. 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.

**Example 1:**

Input: s = "egg", t = "add"  
Output: true

**Example 2:**

Input: s = "foo", t = "bar"  
Output: false

**Example 3:**

Input: s = "paper", t = "title"  
Output: true

**解** 确保s->t和t->s都是单射（一一映射），用两个map分别表示正向和反向的映射关系

class Solution {  
public:  
 bool isIsomorphic(string s, string t) {  
 if(s.size() != t.size())return false;  
 unordered\_map<char, char>mp1, mp2;  
 for(int i = 0; i < s.size(); ++i){  
 if(mp1[t[i]] && mp1[t[i]] != s[i])return false;  
 if(mp2[s[i]] && mp2[s[i]] != t[i])return false;  
 mp1[t[i]] = s[i];  
 mp2[s[i]] = t[i];  
 }  
 return true;  
 }  
};