

Isomorphic Strings

For this problem we must check if characters in s can be mapped one-to-one to characters in t while preserving order.

Key Idea:

We need a bijective mapping:

- Each character in s maps to exactly one character in t .
- No two characters in s map to the same character in t .

Approach: Two Hash Maps

Steps:

1. Create two maps:

· map ST : map $s[i] \rightarrow t[i]$

· map TS : map $t[i] \rightarrow s[i]$ (to ensure bijection)

2. Iterate through both strings simultaneously:

· If mapping exists, check consistency.

· If no mapping exists, create one.

3. If any inconsistency occurs \rightarrow return false.

4. Otherwise, return true.

Complexity:

· Time: $O(n)$ where n = length of strings

· Space: $O(1)$ (limited by ASCII character set size)