30 Days Leetcode challenge - Day 25

public class leetday25 {

class Solution1 {

public String longestCommonPrefix(String[] strs)

{

if (strs == null || strs.length == 0) return "";

String prefix = strs[0];

for (int i = 1; i < strs.length; i++) {

while (strs[i].indexOf(prefix) != 0) {

prefix = prefix.substring(0, prefix.length() - 1);

if (prefix.isEmpty()) return "";

}

}

return prefix;

}

}

class Solution2 {

public boolean isIsomorphic(String s, String t)

{

if (s.length() != t.length()) {

return false;

}

int[] mapS = new int[256];

int[] mapT = new int[256];

for (int i = 0; i < s.length(); i++) {

char charS = s.charAt(i);

char charT = t.charAt(i);

if (mapS[charS] == 0) {

mapS[charS] = charT;

} else {

if (mapS[charS] != charT) {

return false;

}

}

if (mapT[charT] == 0) {

mapT[charT] = charS;

} else {

if (mapT[charT] != charS) {

return false;

}

}

if (mapS[charS] != charT || mapT[charT] != charS) {

return false;

}

}

return true;

}

}

}