/\*

Given a string, find the first non-repeating character in it and return it's index. If it doesn't exist, return -1.

\*/

class Node{

int index,cnt;

Node(int index,int cnt){

this.index=index;

this.cnt=cnt;

}

}

class Solution {

public int firstUniqChar(String s) {

LinkedHashMap <Character,Node> map=new LinkedHashMap<>();

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

char x=s.charAt(i);

if(!map.containsKey(x)){

Node temp=new Node(i,1);

map.put(x,temp);

}

else{

Node temp=map.get(x);

temp.index=i;

temp.cnt=temp.cnt+1;

map.replace(x,map.get(x),temp);

} }

for(Map.Entry<Character,Node> ent:map.entrySet()){

Node temp=ent.getValue();

if(temp.cnt==1)

return temp.index;

} return -1; }}