剑指Offerxx\_字符流中第一个不重复的字符

# 剑指Offerxx\_字符流中第一个不重复的字符

## 题目描述

\* 请实现一个函数用来找出字符流中第一个只出现一次的字符。

\* 例如，当从字符流中只读出前两个字符"go"时，第一个只出现一次的字符是"g"。

\* 当从该字符流中读出前六个字符“google"时，第一个只出现一次的字符是"l"。

\* 输出描述:

\* 如果当前字符流没有存在出现一次的字符，返回#字符。

## 思路分析

\* 思路分析：利用长度为256的哈希表实现;下标对应字符char,可以在O(1)时间内找到该字符对应的出现记录，

\* value为出现记录;未出现默认为-1，出现一次设置其值为索引index;出现两次或多次，设置为-2.

\* Insert方法就是对应上面的记录进行改变。

\* FirstAppearingOnce方法实现要谨慎，每次都需要遍历一遍，因为哈希表是以字符ch为索引，

\* 不是按照出现顺序，而value是index，代表出现顺序，因此首先锁定index>0,表示出现一次，

\* 找出index的最小值，表示最早出现，同时将结果ch设置为此时的数组索引i，最终返回ch即可。

\*/

## Java代码

**private int[] hashTable = new int[256];**

{//初始化hashTable

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

hashTable[i] = -1;

}

private int index = 0;

//Insert one char from stringstream

public void Insert(char ch){

if(hashTable[ch]== -1) hashTable[ch] = index;//若未出现过，则设为index

else if(hashTable[ch] > -1) hashTable[ch] = -2;//若出现过1次，则设为-2

index++;

}

//return the first appearence once char in current stringstream

public char FirstAppearingOnce(){

int minIndex = 256;

char ch = '#';

for(int i = 0;i < 256;i++){

if(hashTable[i] >= 0&&hashTable[i] < minIndex) {//只出现一次,且比上一个出现一次的更早

**minIndex = hashTable[i];**

**ch = (char)i;**

}

}

return ch;

}