

[LeetCode] Palindrome Number & Valid Palindrome - 回文系列问题

原创

Eastmount

2015-09-24 02:48:07

2446

☆ 收藏

版权

分类专栏:

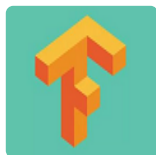
LeetCode

文章标签:

leetcode

数字回文

字符串回文



Python+TensorFlow人工智能

该专栏为人工智能入门专栏，采用Python3和TensorFlow实现人工智能相关算法。前期介绍安装流程、基础语法、神...



Eastmount

¥9.90

订阅博主

题目概述:

Determine whether an integer is a palindrome. Do this without extra space.

题目分析:

判断数字是否是回文 例如121、656、3443

方法有很多，正着看和到着看两数相同；当然负数显然不是回文

我的方法:

第一种方法:

由于没有看到前面的without extra space。采用的方法是把数字转换为字符串，依次比较最前和最后两个字符是否相同，直到遍历完毕。

```
/**
 * 判断一个数字是否是回文数字 Palindrome
 * 最简单方法数组存储数字每位 判断回文 或依次比较数字left和right
 */
bool isPalindrome(int x) {
    char str[20];    // 存储数字位数
    int i,j;
    int count;       // 数字位数

    // 负数肯定不是回文
    if(x<0) return false;

    i=0;
    while(x>0) {
```

```

        str[i]=x%10+'0';          i++;
        x=x/10;
    }

    count=i;
    i=0;
    j=count-1;
    while(i<j) {
        if(str[i]==str[j]) {
            i++;
            j--;
        }
        else {
            return false;
        }
    }
    return true;
}

```

第二种方法：

通过依次比较最高位和最低位数字，然后依次数字降低两位。要点是先计算最高位是多少位。需要注意计算高位时： $x=x-(left*result)$

例如110011 高位-1 低位-1 $110011-100000=10011$ $10011/10=1001$ 再判断，而%会出现错误。

```

/**
 * 判断一个数字是否是回文数字 Palindrome
 * without extra space.
 */
bool isPalindrome(int x) {
    int left,right;    //数字高低位
    int result;        //计算高位数字
    int number;

    if(x<0) return false;

    //计算最高位
    result=1;
    number=x;
    while(number>=10) {    //注意=10
        result=result*10;
        number=number/10;
    }
}

```

```

while(x>0) {
    left=x/result;    //高位
    right=x%10;       //低位
    if(left!=right) {
        return false;
    }
    else { //同时缩小两位 5005-5*1000=5
        x=x-(left*result);    //注意不能是 x=x%(left*result) 1100110011
Runtime Error
        result=result/100;    x=x/10;
    }
}
return true;
}

```

Valid Palindrome

Given a string, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.

For example,

"A man, a plan, a canal: Panama" is a palindrome.

"race a car" is not a palindrome.

该题目主要是判断字符串是否是回文，其中仅仅是字母和数字组成，需要删除额外字符。注意：大小写字母可以相同'z'='Z'，大写字母转小写字母加32即可。

判读一个字符串是否是回文，一种方法可以将字符串倒置然后和原字符串进行比较。这里采用一种类似字符串翻转的方法，通过从前后两个方向来比较判断是否是回文。

自己AC感觉还行，代码如下：

```

//自定义函数 判断是否是数字或字母
bool IsNumberAlp(char ch)
{
    if( (ch<='Z'&&ch>='A') || (ch<='z'&&ch>='a') ||
        (ch<='9'&&ch>='0') ) {
        return true;
    }
    else {
        return false;
    }
}

//字符串回文判断 包括字母和数字 "aA" 返回True说明不区分大小写
bool isPalindrome(char* s) {

```

```

int length;          //字符串长度
int i,j;

char left,right; //比较两个字符

//s输出true ""
if(s==NULL)
    return true;
length=strlen(s);

//数组下标i从前往后遍历 j从后往前遍历
i=0;
j=length-1;
left=0;
right=0;
while(i<=j)
{
    if(IsNumberAlp(s[i])==true) { //字母 数字
        left = s[i];
        if(left>='A'&&left<='Z') { //大写字母转小写字母加32(重点)
            left=left+32;
        }
    }
    else {
        i++;
        continue;
    }

    if(IsNumberAlp(s[j])==true) {
        right = s[j];
        if(right>='A'&&right<='Z') {
            right=right+32;
        }
    }
    else {
        j--;
        continue;
    }

    if(left!=right) {
        return false;
    } else {
        i++;
        j--;
    }
}
return true;
}

```

最后希望文章对你有所帮助，如果文章中有错误或不足之处，还请海涵~
(By:Eastmount 2015-9-24 凌晨4点 <http://blog.csdn.net/eastmount/>)