

# Assign #3: Oct Mock Exam暨选做题目满百

Updated 1537 GMT+8 Oct 10, 2024

2024 fall, Compiled by 俞天麒 物理学院

## 说明:

- 1) Oct月考: AC6== (请改为同学的通过数) == 。考试题目都在“题库 (包括计概、数算题目)”里面, 按照数字题号能找到, 可以重新提交。作业中提交自己最满意版本的代码和截图。
- 2) 请把每个题目解题思路 (可选), 源码Python, 或者C++/C (已经在Codeforces/Openjudge上AC), 截图 (包含Accepted, 学号), 填写到下面作业模版中 (推荐使用 typora <https://typoraio.cn>, 或者用word)。AC 或者没有AC, 都请标上每个题目大致花费时间。
- 3) 提交时候先提交pdf文件, 再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、作业评论有md或者doc。
- 4) 如果不能在截止前提交作业, 请写明原因。

## 1. 题目

### E28674: 《黑神话: 悟空》之加密

<http://cs101.openjudge.cn/practice/28674/>

思路:

忘了怎么让字母顺序加k了, 于是直接打表 (滚去复习了

代码

```
k=int(input())
a=input()
ans=[]
down=
["a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q","r","s","t",
"u","v","w","x","y","z"]
up=
["A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P","Q","R","S","T",
"U","V","W","X","Y","Z"]
for i in range(len(a)):
    if a[i] in set(down):
        ans.append(down[(down.index(a[i])-k)%26])
    elif a[i] in set(up):
        ans.append(up[(up.index(a[i]) - k) % 26])
print("".join(ans))
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
k=int(input())
a=input()
ans=[]
down=["a","b","c","d","e","f","g","h","i","j","k","l","m","n","o","p","q","r","s"]
up=["A","B","C","D","E","F","G","H","I","J","K","L","M","N","O","P","Q","R","S"]
for i in range(len(a)):
    if a[i] in set(down):
        ans.append(down[(down.index(a[i])-k)%26])
    elif a[i] in set(up):
        ans.append(up[(up.index(a[i]) - k) % 26])
print("".join(ans))
```

基本信息

#: 46405849  
题目: 28674  
提交人: 24n2400011425  
内存: 3664kB  
时间: 22ms  
语言: Python3  
提交时间: 2024-10-10 17:06:09

## E28691: 字符串中的整数求和

<http://cs101.openjudge.cn/practice/28691/>

思路:

利用int把字符串前两个字符转化成整数

代码

```
a,b=input().split()
print(int(a[0:2])+int(b[0:2]))
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
a,b=input().split()
print(int(a[0:2])+int(b[0:2]))
```

基本信息

#: 46405916  
题目: 28691  
提交人: 24n2400011425  
内存: 3588kB  
时间: 21ms  
语言: Python3  
提交时间: 2024-10-10 17:07:05

## M28664: 验证身份证号

<http://cs101.openjudge.cn/practice/28664/>

思路:

题目怎么说怎么来，可以利用列表索引来简化分支结构

代码

```

correct=["1","0","x","9","8","7","6","5","4","3","2"]
multiply=[7,9,10,5,8,4,2,1,6,3,7,9,10,5,8,4,2]
def judge(num,a):
    if a==correct[num]:
        return True
    else:
        return False
n=int(input())
for _ in range(n):
    a=input()
    ans=0
    for i in range(17):
        ans+=int(a[i])*multiply[i]
    if judge(ans%11,a[17])==True:
        print("YES")
    else:
        print("NO")

```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: **Accepted**

源代码

```

correct=["1","0","x","9","8","7","6","5","4","3","2"]
multiply=[7,9,10,5,8,4,2,1,6,3,7,9,10,5,8,4,2]
def judge(num,a):
    if a==correct[num]:
        return True
    else:
        return False
n=int(input())
for _ in range(n):
    a=input()
    ans=0
    for i in range(17):
        ans+=int(a[i])*multiply[i]
    if judge(ans%11,a[17])==True:
        print("YES")
    else:
        print("NO")

```

基本信息

#: 46405953  
 题目: 28664  
 提交人: 24n2400011425  
 内存: 3596kB  
 时间: 21ms  
 语言: Python3  
 提交时间: 2024-10-10 17:07:46

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M28678: 角谷猜想

<http://cs101.openjudge.cn/practice/28678/>

思路:

题目怎么说怎么来

代码

```

n=int(input())
while n!=1:
    if n%2==0:
        print(str(n)+"/2="+str(int(n/2)))
        n=n//2
    else:
        print(str(n)+"*3+1="+str(n*3+1))
        n=n*3+1
print("End")

```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: Accepted

源代码

```

n=int(input())
while n!=1:
    if n%2==0:
        print(str(n)+"/2="+str(int(n/2)))
        n=n//2
    else:
        print(str(n)+"*3+1="+str(n*3+1))
        n=n*3+1
print("End")

```

基本信息

#: 46406040  
 题目: 28678  
 提交人: 24n2400011425  
 内存: 3600kB  
 时间: 21ms  
 语言: Python3  
 提交时间: 2024-10-10 17:08:56

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M28700: 罗马数字与整数的转换

<http://cs101.openjudge.cn/practice/28700/>

思路:

两个部分, 从罗马数字转回时就按照每个字母判断来作加法, 从整数转过去把他按照位数分开, 从高位往低位用分支结构处理 (我这里做复杂了, 应该按照1000, 500, 100, 50这样来分更简单, 但是考场上就将错就错了)

代码:

```

a=input()
letter=["I","V","X","L","C","D","M"]
respond=[1,5,10,50,100,500,1000]
if a[0] in set(letter):
    a=list(a)
    ans=0
    i=0
    while len(a)>1:
        r=1
        if a[0]=="M":
            ans+=1000
        elif a[0]=="D":
            ans+=500
        elif a[0]=="L":

```

```

        ans+=50
    elif a[0]=="V":
        ans+=5
    elif a[0]=="C":
        if a[1]=="M":
            ans+=900
            r=2
        elif a[1]=="D":
            ans+=400
            r=2
        else:
            ans+=100
    elif a[0]=="X":
        if a[1]=="C":
            ans+=90
            r=2
        elif a[1]=="L":
            ans+=40
            r=2
        else:
            ans+=10
    elif a[0]=="I":
        if a[1]=="X":
            ans+=9
            r=2
        elif a[1]=="V":
            ans+=4
            r=2
        else:
            ans+=1
    if r==1:
        a.pop(0)
    else:
        a.pop(0)
        a.pop(0)
    if len(a)>0:
        ans+=respond[letter.index(a[0])]
    print(ans)
else:
    a=int(a)
    thousand=a//1000
    hundred=(a-thousand*1000)//100
    ten=(a-thousand*1000-hundred*100)//10
    one=a-thousand*1000-hundred*100-ten*10
    if thousand>0:
        for i in range(thousand):
            print("M",end="")
    if hundred>0:
        if hundred==9:
            print("CM",end="")
        elif hundred==4:
            print("CD",end="")
        else:
            if hundred>=5:
                print("D",end="")
                hundred-=5

```

```

        for i in range(hundred):
            print("C",end="")
if ten>0:
    if ten==9:
        print("XC",end="")
    elif ten==4:
        print("XL",end="")
    else:
        if ten>=5:
            print("L",end="")
            ten-=5
        for i in range(ten):
            print("X",end="")
if one>0:
    if one==9:
        print("IX",end="")
    elif one==4:
        print("IV",end="")
    else:
        if one>=5:
            print("V",end="")
            one-=5
        for i in range(one):
            print("I",end="")

```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==

状态: Accepted

源代码

```

from traceback import print_tb

a=input()
letter=["I","V","X","L","C","D","M"]
respond=[1,5,10,50,100,500,1000]
if a[0] in set(letter):
    a=list(a)
    ans=0
    i=0
    while len(a)>1:
        r=1
        if a[0]=="M":
            ans+=1000
        elif a[0]=="D":
            ans+=500
        else:
            ans+=respond[i]
            i+=1
        a.pop(0)
    print(ans)

```

基本信息

# : 46406158  
题目: 28700  
提交人: 24n2400011425  
内存: 4068kB  
时间: 25ms  
语言: Python3  
提交时间: 2024-10-10 17:10:44

## \*T25353: 排队 (选做)

<http://cs101.openjudge.cn/practice/25353/>

思路:

一开始我想说去搜索 $n$ 次每次找出能放到左边的最小的元素然后挪过去，但是这样子时间复杂度是 $O(n^3)$ ，之后我想说用一个 $n*n$ 的列表来储存每一个数据与前面的数据是否可交换的信息，但是这样子储存空间要 $n^2$ ，时间复杂度还是 $O(n^2)$ ，还是过不了。之后看到了<https://www.cnblogs.com/guoshayang/p/17824372.html>说的“层数”的思想，于是想在输入时同时遍历一遍之前的数据来看看层数是多少，但是这样子复杂度还是 $O(n^2)$ ，还是过不了。看到标准答案里用树，打算等到算法部分好好学了再回头看，下面是我目前能想到的最简单的算法。

代码

```
n,d=map(int,input().split())
a=[]
for i in range(n):
    x=int(input())
    rank=0
    for j in range(i):
        if abs(a[j][0]-x)>d:
            rank=max(rank,a[j][1]+1)
    a.append([x,rank])
a.sort(key=lambda x:(x[1],x[0]))
for i in range(n):
    print(a[i][0])
```

代码运行截图 == (AC代码截图，至少包含有"Accepted") ==

超时。。。

## 2. 学习总结和收获

==如果作业题目简单，有否额外练习题目，比如：OJ“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。==

这次考试前面5题都很简单，最后一题应该是我自己优化没想到+算法基本没学过。之后还得往这两个方向下功夫，多见一些题目看看多样化的优化思路，以及之后的课好好听学点算法。希望期末上机考试别有最后一题这么难的东西。。。