

# Assignment #7: Nov Mock Exam立冬

Updated 1646 GMT+8 Nov 7, 2024

2024 fall, Compiled by 俞天麒 物理学院

## 说明:

- 1) 月考: AC6。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++(已经在Codeforces/Openjudge上AC),截图(包含Accepted),填写到下面作业模版中(推荐使用typora <https://typora.io.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”区有上传的md或者doc附件。
- 4) 如果不能在截止前提交作业,请写明原因。

## 1. 题目

### E07618: 病人排队

sortings, <http://cs101.openjudge.cn/practice/07618/>

思路:

在储存时分成两个数组,一个储存老人,另一个是其他,然后分别输出就好

代码:

```
n=int(input())
a=[]
b=[]
for i in range(n):
    num,age=input().split()
    age=int(age)
    if age>=60:
        b.append([num,age,n-i])
    else:
        a.append([num,age])
b.sort(key=lambda x:(x[1],x[2]),reverse=True)
for i in range(len(b)):
    print(b[i][0])
for i in range(len(a)):
    print(a[i][0])
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n=int(input())
a=[]
b=[]
for i in range(n):
    num,age=input().split()
    age=int(age)
    if age>=60:
        b.append([num,age,n-i])
    else:
        a.append([num,age])
b.sort(key=lambda x:(x[1],x[2]),reverse=True)
for i in range(len(b)):
    print(b[i][0])
for i in range(len(a)):
    print(a[i][0])
```

基本信息

#: 47009777  
题目: E07618  
提交人: 24n2400011425  
内存: 3640kB  
时间: 24ms  
语言: Python3  
提交时间: 2024-11-07 15:15:17

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## E23555: 节省存储的矩阵乘法

implementation, matrices, <http://cs101.openjudge.cn/practice/23555/>

思路:

4层循环硬遍历, 但是居然没超时

代码:

```
n,m1,m2=map(int,input().split())
a=[]
b=[]
for i in range(m1):
    a.append(list(map(int,input().split())))
for i in range(m2):
    b.append(list(map(int,input().split())))
for i in range(n):
    for j in range(n):
        ans=0
        for k in range(m1):
            if a[k][0]==i:
                for l in range(m2):
                    if b[l][1]==j and b[l][0]==a[k][1]:
                        ans+=a[k][2]*b[l][2]
        if ans!=0:
            print(i,j,ans)
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
n,m1,m2=map(int,input().split())
a=[]
b=[]
for i in range(m1):
    a.append(list(map(int,input().split())))
for i in range(m2):
    b.append(list(map(int,input().split())))
for i in range(n):
    for j in range(n):
        ans=0
        for k in range(m1):
            if a[k][0]==i:
                for l in range(m2):
                    if b[l][1]==j and b[l][0]==a[k][1]:
                        ans+=a[k][2]*b[l][2]
        if ans!=0:
            print(i,j,ans)
```

基本信息

#: 47010012  
题目: E23555  
提交人: 24n2400011425  
内存: 3792kB  
时间: 531ms  
语言: Python3  
提交时间: 2024-11-07 15:22:43

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M18182: 打怪兽

implementation/sortings/data structures, <http://cs101.openjudge.cn/practice/18182/>

思路:

按照同一个时刻伤害从高到低排序, 然后优先输出同一时刻的前m个。故用一个数mark标记在同一个时刻已经用了几个武器, 当切换时刻的时候把mark重置一下。

代码:

```
t=int(input())
for _ in range(t):
    n,m,b=map(int,input().split())
    a=[]
    for i in range(n):
        a.append(list(map(int,input().split())))
    a.sort(key=lambda x:(x[0],-x[1]))
    i=1
    mark=1
    b-=a[0][1]
    if b<=0:
        print(a[0][0])
        continue
    for i in range(1,n):
        if a[i][0]!=a[i-1][0]:
            mark=1
            b-=a[i][1]
        else:
            if mark<m:
                mark+=1
                b-=a[i][1]
            if b<=0:
                print(a[i][0])
                break
    if b>0:
        print("alive")
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
t=int(input())
for _ in range(t):
    n,m,b=map(int,input().split())
    a=[]
    for i in range(n):
        a.append(list(map(int,input().split())))
    a.sort(key=lambda x:(x[0],-x[1]))
    i=1
    mark=1
    b-=a[0][1]
    if b<=0:
        print(a[0][0])
        continue
    for i in range(1,n):
        if a[i][0]!=a[i-1][0]:
            mark=1
            b-=a[i][1]
        else:
            if mark<m:
                mark+=1
                b-=a[i][1]
        if b<=0:
            print(a[i][0])
            break
    if b>0:
        print("alive")
```

基本信息

#: 47010740  
题目: M18182  
提交人: 24n2400011425  
内存: 3856kB  
时间: 79ms  
语言: Python3  
提交时间: 2024-11-07 15:39:02

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#)

## M28780: 零钱兑换3

dp, <http://cs101.openjudge.cn/practice/28780/>

思路:

完全背包, 套模板

代码:

```
n,m=map(int,input().split())
ans=[float("inf")]*(m+1)
ans[0]=0
a=list(map(int,input().split()))
for i in range(n):
    for j in range(a[i],m+1):
        ans[j]=min(ans[j],ans[j-a[i]]+1)
    #print(ans)
if ans[m]==float("inf"):
    print(-1)
else:
    print(ans[m])
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n,m=map(int,input().split())
ans=[float("inf")]*(m+1)
ans[0]=0
a=list(map(int,input().split()))
for i in range(n):
    for j in range(a[i],m+1):
        ans[j]=min(ans[j],ans[j-a[i]]+1)
    #print(ans)
if ans[m]==float("inf"):
    print(-1)
else:
    print(ans[m])
```

基本信息

#: 47010961  
题目: M28780  
提交人: 24n2400011425  
内存: 28640kB  
时间: 15423ms  
语言: Python3  
提交时间: 2024-11-07 15:44:11

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## T12757: 阿尔法星人翻译官

implementation, <http://cs101.openjudge.cn/practice/12757>

思路:

最最最恶心的一题，首先上来打那个字典先打3min，然后还要注意其实英语的逻辑有点类似递归，处理thousand量级的时候要分thousand前后，分别读一次，前面的乘1000再加后面，million的时候也是找到million，对前面后面分别调用一次thousand的那个读法，然后加起来，所以定义了三个函数分别处理thousand以下，thousand，million。

代码:

```
dic={"zero":0,"one":1,"two":2,"three":3,"four":4,"five":5,"six":6,"seven":7,
"eight":8,"nine":9,"ten":10,"eleven":11,"twelve":12,"thirteen":13,
"fourteen":14,"fifteen":15,"sixteen":16,"seventeen":17,"eighteen":18,
"nineteen":19,"twenty":20,"thirty":30,"forty":40,"fifty":50,
"sixty":60,"seventy":70,"eighty":80,"ninety":90,"hundred":100,
"thousand":1000,"million":1000000}
a=list(input().split())
def find(x):
    ans=0
    i=0
    while i<len(x):
        if x[i]=="hundred":
            ans*=100
        else:
            ans+=dic[x[i]]
        i+=1
    return ans
def thou(x):
    if "thousand" in x:
        ind=x.index("thousand")
        return find(x[0:ind])*1000+find(x[ind+1:])
    else:
        return find(x)
def mill(x):
    if "million" in x:
        ind=x.index("million")
        return thou(x[0:ind])*1000000+thou(x[ind+1:])
    else:
```

```

        return thou(x)
mark,i=False,0
if a[0]=="negative":
    mark=True
    i=1
ans=mill(a[i::])
if mark==False:
    print(ans)
else:
    print(-ans)

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

dic={"zero":0,"one":1,"two":2,"three":3, "four":4, "five":5, "six":6, "seven":7}
a=list(input().split())
def find(x):
    ans=0
    i=0
    while i<len(x):
        if x[i]=="hundred":
            ans*=100
        else:
            ans+=dic[x[i]]
        i+=1
    return ans
def thou(x):
    if "thousand" in x:
        ind=x.index("thousand")
        return find(x[0:ind])*1000+find(x[ind+1::])
    else:
        return find(x)
def mill(x):
    if "million" in x:
        ind=x.index("million")
        return thou(x[0:ind])*1000000+thou(x[ind+1::])
    else:
        return thou(x)
mark,i=False,0
if a[0]=="negative":
    mark=True
    i=1
ans=mill(a[i::])
if mark==False:
    print(ans)
else:
    print(-ans)

```

基本信息

#: 47012541  
 题目: T12757  
 提交人: 24n2400011425  
 内存: 3720kB  
 时间: 25ms  
 语言: Python3  
 提交时间: 2024-11-07 16:16:14

## T16528: 充实的寒假生活

greedy/dp, cs10117 Final Exam, <http://cs101.openjudge.cn/practice/16528/>

思路:

区间覆盖, 问最多有几个不重叠的区间, 套模板

代码:

```
n=int(input())
a=[]
for i in range(n):
    a.append(list(map(int,input().split())))
ans=1
a.sort(key=lambda x:x[1])
mark=a[0][1]
for i in range(1,n):
    if a[i][0]>mark:
        ans+=1
        mark=a[i][1]
print(ans)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n=int(input())
a=[]
for i in range(n):
    a.append(list(map(int,input().split())))
ans=1
a.sort(key=lambda x:x[1])
mark=a[0][1]
for i in range(1,n):
    if a[i][0]>mark:
        ans+=1
        mark=a[i][1]
print(ans)
```

基本信息

#: 47012808  
题目: T16528  
提交人: 24n2400011425  
内存: 4076kB  
时间: 36ms  
语言: Python3  
提交时间: 2024-11-07 16:20:04

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## 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。

这次考试做起来挺顺的，可能是因为每日选做都有跟进，也因此积累了许多模板，比如各种背包，区间之类的，刚好能对上这次考试的几个模板题，但是不是模板的题目做起来还是有点费力，希望下次能多动脑子，解决好没见过的题目。