

Assignment #7: Nov Mock Exam 立冬

Updated 1646 GMT+8 Nov 7, 2024

2024 fall, Compiled by <mark>祁黄奕 城市与环境学院</mark>

****说明: ****

1) 月考: AC6<mark>AC4</mark>。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://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”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

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

思路: 分别记录病人的 ID 和年龄, 建立列表储存数组数据, ≥ 60 岁的先按年龄大小排序, 确保同年龄的按照输入先后顺序; 其余的按照输入先后顺序排好即可。注意为了不超时, sort 时只需考虑必要项。

代码:

```
```python
```

```

n=int(input())
p=[]
for i in range(n):
 id,age=input().split()
 age=int(age)
 p.append((id,age,i))
p.sort(key=lambda p: (-p[1],p[2]) if p[1]>=60 else (p[2],))
for ppl in p:
 print(ppl[0])
```

```

代码运行截图 <mark>（至少包含有“Accepted”）</mark>

#47072336提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n=int(input())
p=[]
for i in range(n):
    id,age=input().split()
    age=int(age)
    p.append((id,age,i))
p.sort(key=lambda p: (-p[1],p[2]) if p[1]>=60 else (p[2],))
for ppl in p:
    print(ppl[0])

```

基本信息

#: 47072336
 题目: 07618
 提交人: qhy
 内存: 3632kB
 时间: 25ms
 语言: Python3
 提交时间: 2024-11-10 11:59:59

©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：先储存两个矩阵的数据，按照公式求出乘积后分行打印。这题我没思路所以是看的题解，原理不难但是一直不会矩阵运算，吃亏了。

代码：

```python

```

n, m1, m2 = map(int, input().split())
A = {}
B = {}
C = {}
for _ in range(m1):
 x, y, v = map(int, input().split())
 A[(x, y)] = v
for _ in range(m2):
 x, y, v = map(int, input().split())
 B[(x, y)] = v

for i in range(n):
 for k in range(n):
 s = sum(A.get((i, j), 0)*B.get((j, k), 0) for j in
range(n))
 if s != 0:

```

```

 C[(i, k)] = s

ans = sorted(((k[0], k[1], v) for k, v in C.items()))
for x in ans:
 print(*x)
```

```

代码运行截图 ==（至少包含有“Accepted”）==

#47072682提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n, m1, m2 = map(int, input().split())
A = {}
B = {}
C = {}
for _ in range(m1):
    x, y, v = map(int, input().split())
    A[(x, y)] = v
for _ in range(m2):
    x, y, v = map(int, input().split())
    B[(x, y)] = v

for i in range(n):
    for k in range(n):
        s = sum(A.get((i, j), 0)*B.get((j, k), 0) for j in range(n))
        if s != 0:
            C[(i, k)] = s

ans = sorted(((k[0], k[1], v) for k, v in C.items()))
for x in ans:
    print(*x)

```

基本信息

#: 47072682
 题目: 23555
 提交人: qhy
 内存: 4064kB
 时间: 35ms
 语言: Python3
 提交时间: 2024-11-10 12:26:08

©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： 计算取同一时间最大的 m 个技能和。这题自己写的代码老是 TLE，考试时没有通过。最后实在受不了了看题解也没完全看懂，姑且先交着吧。

代码：

```python

```

for _ in range(int(input())):
 n, m, b = map(int, input().split())
 s = {}
 ts = []
 for _ in range(n):
 t, x = map(int, input().split())
 if t in s:
 s[t].append(x)
 else:
 s[t] = [x]
 ts.append(t)

```

```

ts.sort()
for t in ts:
 d = sum(sorted(s[t], reverse=True)[:m])
 b -= d
 if b <= 0:
 print(t)
 break
if b > 0:
 print("alive")

```

代码运行截图 <mark>（至少包含有“Accepted”）</mark>

#47076327提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

for _ in range(int(input())):
 n, m, b = map(int, input().split())
 s = {}
 ts = []
 for _ in range(n):
 t, x = map(int, input().split())
 if t in s:
 s[t].append(x)
 else:
 s[t] = [x]
 ts.append(t)
 ts.sort()
 for t in ts:
 d = sum(sorted(s[t], reverse=True)[:m])
 b -= d
 if b <= 0:
 print(t)
 break
 if b > 0:
 print("alive")

```

基本信息

#: 47076327  
 题目: 18182  
 提交人: qhy  
 内存: 4116kB  
 时间: 76ms  
 语言: Python3  
 提交时间: 2024-11-10 15:24:46

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

### M28780: 零钱兑换 3

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

思路: 建立 dp 列表, 如果当前金额  $i$  减去当前硬币面值  $coin$  后仍然大于等于 0, 则更新  $dp[i]$  为  $dp[i - coin] + 1$  和  $dp[i]$  中的较小值。这一步的意思是: 若用当前硬币可凑成金额  $i$ , 那么需要的硬币数就是  $dp[i - coin] + 1$ 。

代码:

```python

```

def min_coins(n, m, coins):
    dp = [m + 1] * (m + 1)
    dp[0] = 0

    for i in range(1, m + 1):
        for coin in coins:
            if i - coin >= 0:
                dp[i] = min(dp[i], dp[i - coin] + 1)

    return dp[m] if dp[m] != m + 1 else -1

```

```
n, m = map(int, input().split())
coins = list(map(int, input().split()))
print(min_coins(n, m, coins))
```

```

代码运行截图 <mark>（至少包含有“Accepted”）</mark>

#47077086提交状态

查看 提交 统计 提问

状态: Accepted

源代码

```
def min_coins(n, m, coins):
 dp = [m + 1] * (m + 1)
 dp[0] = 0

 for i in range(1, m + 1):
 for coin in coins:
 if i - coin >= 0:
 dp[i] = min(dp[i], dp[i - coin] + 1)

 return dp[m] if dp[m] != m + 1 else -1

n, m = map(int, input().split())
coins = list(map(int, input().split()))
print(min_coins(n, m, coins))
```

基本信息

#: 47077086  
 题目: 28780  
 提交人: qhy  
 内存: 8596kB  
 时间: 11971ms  
 语言: Python3  
 提交时间: 2024-11-10 15:43:48

©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: 建立映射字典, 注意 100 要乘, 1000、1000000 乘完之后需要 current 归零, 重新进行后续字符转换和计算。

代码:

```python

```
word_to_num = {
    'negative': -1,
    '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
}

def parse_number(words):
    result = 0
    current = 0
    is_negative = False

    for word in words:
        if word == 'negative':
            is_negative = True
        elif word in word_to_num:
            num = word_to_num[word]
            if num == 100:
                current *= num
            elif num == 1000 or num == 1000000:
                result += current * num
                current = 0
            else:
                current += num

    result += current

    if is_negative:
        result = -result

    return result
```

```
input_str = input().strip().split()
result = parse_number(input_str)
print(result)
```
```

代码运行截图 <mark>（至少包含有“Accepted”）</mark>

#47015713提交状态

查看 提交 统计 提问

状态: Accepted

基本信息

#: 47015713  
题目: T12757  
提交人: qhy  
内存: 3672kB  
时间: 28ms  
语言: Python3  
提交时间: 2024-11-07 16:54:27

源代码

```
word_to_num = {
 'negative': -1,
 '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
}

def parse_number(words):
 result = 0
 current = 0
 is_negative = False

 for word in words:
 if word == 'negative':
 is_negative = True
 elif word in word_to_num:
 num = word_to_num[word]
 if num == 100:
 current *= num
 elif num == 1000 or num == 1000000:
 result += current * num
 current = 0
 else:
 current += num

 result += current

 if is_negative:
 result = -result

 return result

input_str = input().strip().split()
result = parse_number(input_str)
print(result)
```

### T16528: 充实的寒假生活

greedy/dp, cs10117 Final Exam,

<http://cs101.openjudge.cn/practice/16528/>

思路: 可参加的活动总和初始为1, 检查新的活动开始时间是否在前一个结束时间之后, 是的话就计算可参加的活动总和时就+1。

代码:

```
```python
```

```

n=int(input())
lst=[]
for _ in range(n):
    begin,end=map(int,input().split())
    lst.append([begin,end])
lst.sort(key=lambda x:x[1])
count=1
last_end=lst[0][1]
for i in range(1,n):
    if lst[i][0]>last_end:
        count+=1
        last_end=lst[i][1]
print(count)

```

代码运行截图 <mark>（至少包含有“Accepted”）</mark>

#47078448提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

n=int(input())
lst=[]
for _ in range(n):
    begin,end=map(int,input().split())
    lst.append([begin,end])
lst.sort(key=lambda x:x[1])
count=1
last_end=lst[0][1]
for i in range(1,n):
    if lst[i][0]>last_end:
        count+=1
        last_end=lst[i][1]
print(count)

```

基本信息

#: 47078448
 题目: 16528
 提交人: qhy
 内存: 3856kB
 时间: 34ms
 语言: Python3
 提交时间: 2024-11-10 16:22:20

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

2. 学习总结和收获

所实话练习还是非常不够，每日选做落下好多，做题时经常有小错误还是一如既往地 debug 不出来。机考借助了 ai，因为实在 debug 不出（非常抱歉）。还是经常有思路但写的不准确导致 WA 或 TLE。。。。。。下次试试完全不用 ai 看看能过多少吧。。。。。。没想到最后一题居然是最简单的，还好习惯性一次把所有题目都点开看了看。期中已经过去，会抓紧跟进编程学习的，希望期末别挂啊啊啊啊啊啊啊啊啊啊啊啊！真的拜托了呃呃呃呃呃！