Assignment #7: Nov Mock Exam立冬

Updated 1646 GMT+8 Nov 7, 2024

说明：

1）月考： AC4（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++（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/

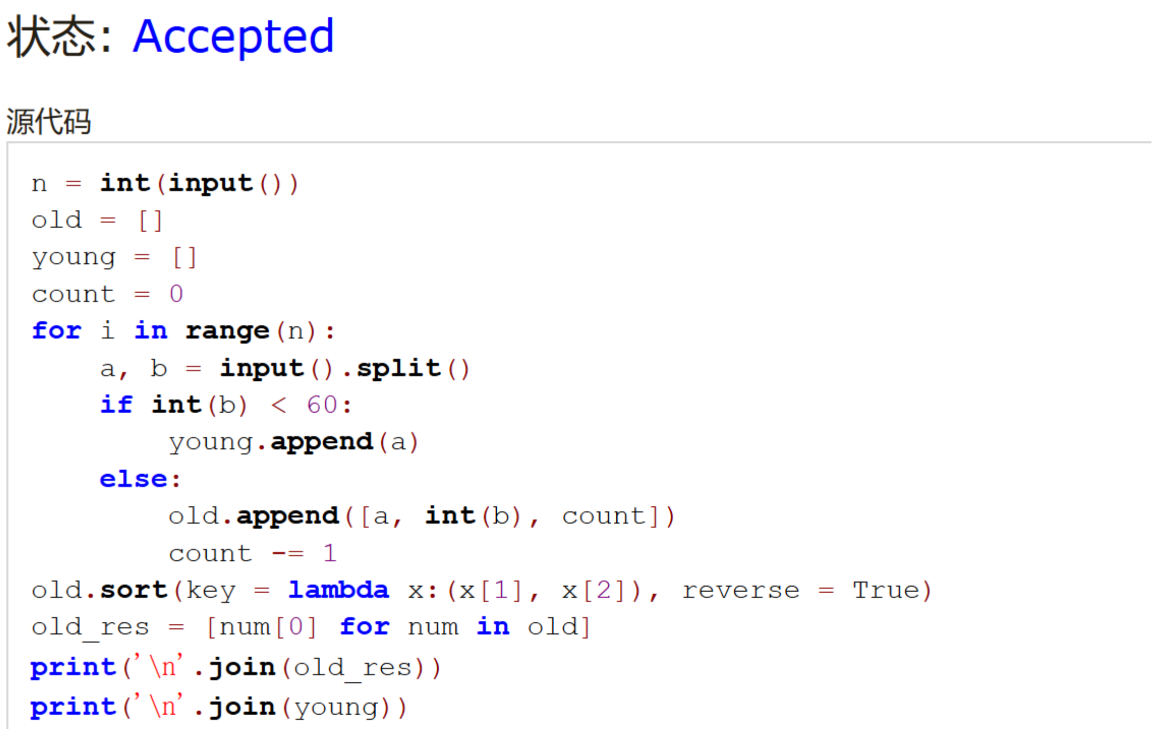
思路：

先把老人和非老人放进两个不同的分组，进行不同门类的排序，因为老人对非老人有绝对优先权，直接分开输出就可以

代码：

```
n = int(input())
old = []
young = []
count = 0
for i in range(n):
    a, b = input().split()
    if int(b) < 60:
        young.append(a)
    else:
        old.append([a, int(b), count])
        count -= 1
old.sort(key = lambda x:(x[1], x[2]), reverse = True)
old_res = [num[0] for num in old]
print('\n'.join(old_res))
print('\n'.join(young))
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
n = int(input())
old = []
young = []
count = 0
for i in range(n):
    a, b = input().split()
    if int(b) < 60:
        young.append(a)
    else:
        old.append([a, int(b), count])
        count -= 1
old.sort(key = lambda x:(x[1], x[2]), reverse = True)
old_res = [num[0] for num in old]
print(’\n’.join(old_res))
print(’\n’.join(young))
```

E23555: 节省存储的矩阵乘法

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

思路：

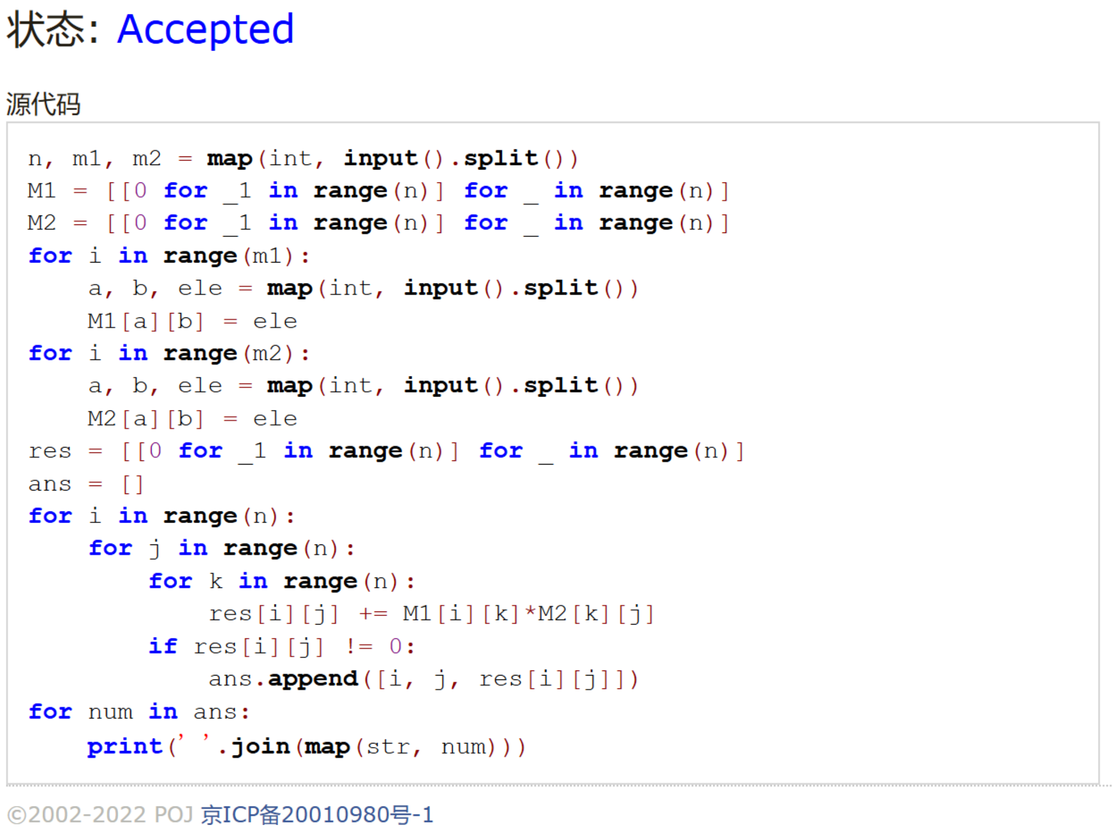
纯笨蛋做法，先还原矩阵再进行乘法计算，计算的时候顺便把非零项先存储下来，最后输出

（有想过能不能先挑选出非零项再计算特定位置的值，但是考试吗先ac再说）

代码：

```python
n, m1, m2 = map(int, input().split())
M1 = [[0 for _1 in range(n)] for _ in range(n)]
M2 = [[0 for _1 in range(n)] for _ in range(n)]
for i in range(m1):
    a, b, ele = map(int, input().split())
    M1[a][b] = ele
for i in range(m2):
    a, b, ele = map(int, input().split())
    M2[a][b] = ele
res = [[0 for _1 in range(n)] for _ in range(n)]
ans = []
for i in range(n):
    for j in range(n):
        for k in range(n):
            res[i][j] += M1[i][k]*M2[k][j]
        if res[i][j] != 0:
            ans.append([i, j, res[i][j]])
for num in ans:
    print(' '.join(map(str, num)))
```

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())
M1 = [[0 for _1 in range(n)] for _ in range(n)]
M2 = [[0 for _1 in range(n)] for _ in range(n)]
for i in range(m1):
    a, b, ele = map(int, input().split())
    M1[a][b] = ele
for i in range(m2):
    a, b, ele = map(int, input().split())
    M2[a][b] = ele
res = [[0 for _1 in range(n)] for _ in range(n)]
ans = []
for i in range(n):
    for j in range(n):
        for k in range(n):
            res[i][j] += M1[i][k]*M2[k][j]
        if res[i][j] != 0:
            ans.append([i, j, res[i][j]])
for num in ans:
    print(' '.join(map(str, num)))
```

©2002-2022 POJ 京ICP备20010980号-1

M18182: 打怪兽

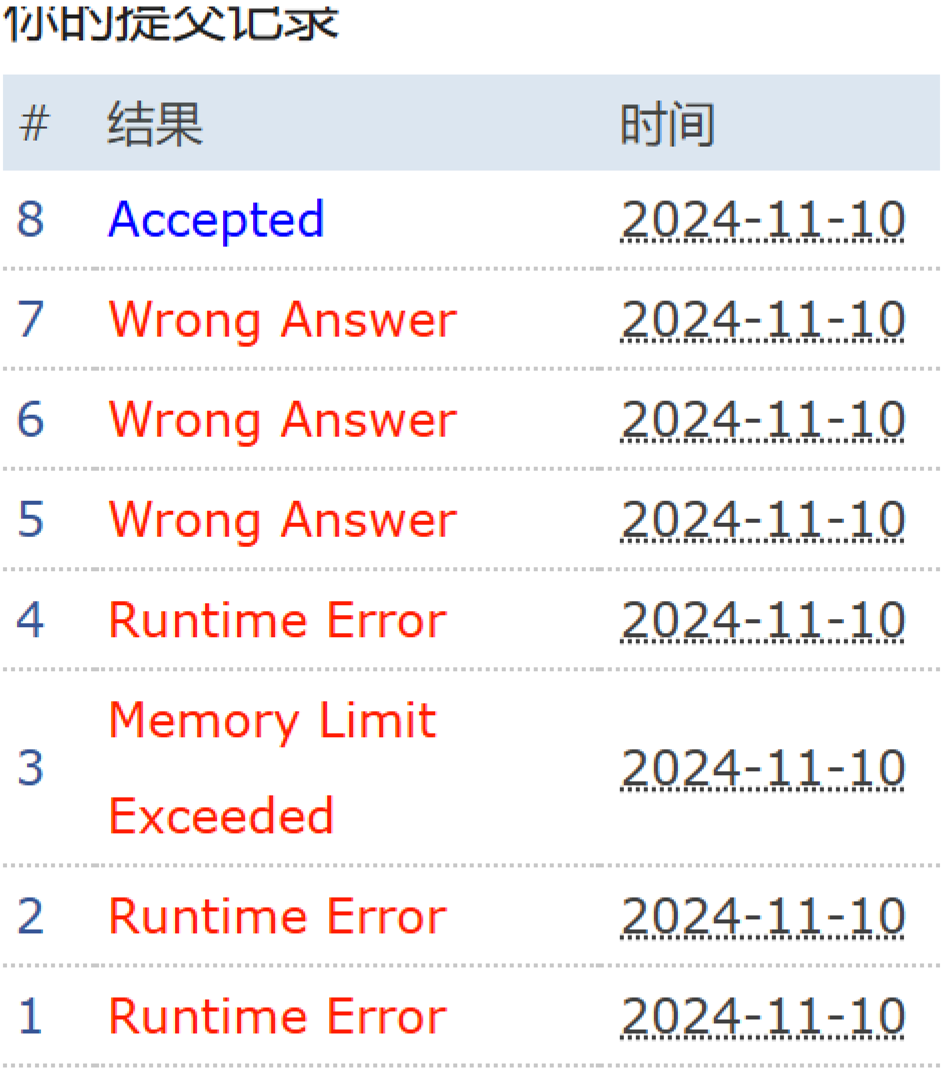
implementation/sortings/data structures, http://cs101.openjudge.cn/practice/18182/

思路：

代码：

```
for _ in range(int(input())):
    n, m, b = map(int, input().split())
    d = {}
    for i in range(n):
        t, x=map(int, input().split())
        if t not in d.keys():
            d[t] = [x]
        else:
            d[t].append(x)
    for i in d.keys():
        d[i].sort(reverse = True)
        d[i] = sum(d[i][:m])
    dp = sorted(d.items())
    for i in dp:
        b -= i[1]
        if b<=0:
            print(i[0])
            break
    else:
        print('alive')
```

代码运行截图 （至少包含有"Accepted"）



你的提交记录

#	结果	时间
8	Accepted	2024-11-10
7	Wrong Answer	2024-11-10
6	Wrong Answer	2024-11-10
5	Wrong Answer	2024-11-10
4	Runtime Error	2024-11-10
3	Memory Limit Exceeded	2024-11-10
2	Runtime Error	2024-11-10
1	Runtime Error	2024-11-10

（好曲折的过程……其实是因为一开始题目理解错了）

M28780: 零钱兑换3

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

思路：

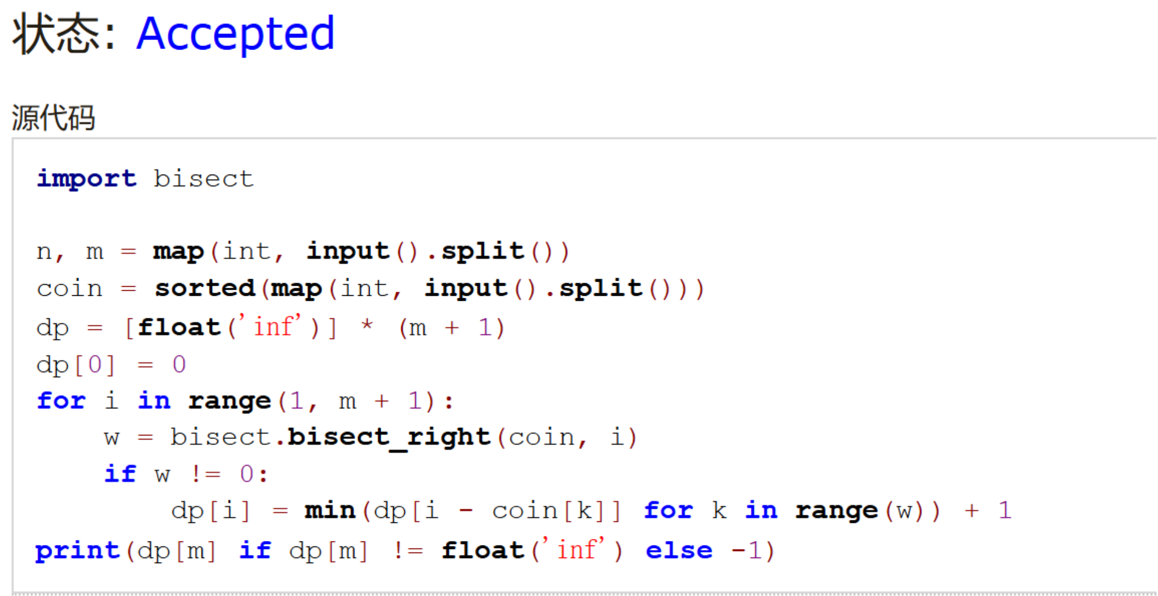
一开始用割绳子的思路写TLE了，之后引入bisect库从反方向来了一遍

代码：

```
import bisect

n, m = map(int, input().split())
coin = sorted(map(int, input().split()))
dp = [float('inf')] * (m + 1)
dp[0] = 0
for i in range(1, m + 1):
    w = bisect.bisect_right(coin, i)
    if w != 0:
        dp[i] = min(dp[i - coin[k]] for k in range(w)) + 1
print(dp[m] if dp[m] != float('inf') else -1)
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
import bisect

n, m = map(int, input().split())
coin = sorted(map(int, input().split()))
dp = [float(’inf’)] * (m + 1)
dp[0] = 0
for i in range(1, m + 1):
    w = bisect.bisect_right(coin, i)
    if w != 0:
        dp[i] = min(dp[i - coin[k]] for k in range(w)) + 1
print(dp[m] if dp[m] != float(’inf’) else -1)
```

T12757: 阿尔法星人翻译官

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

思路：

放在字典里，因为读取的数据小于10**9，方便million的处理，所以找到thousand，million，hundred之后乘以前缀数字相加就可以了（但是打字典手是真疼啊）

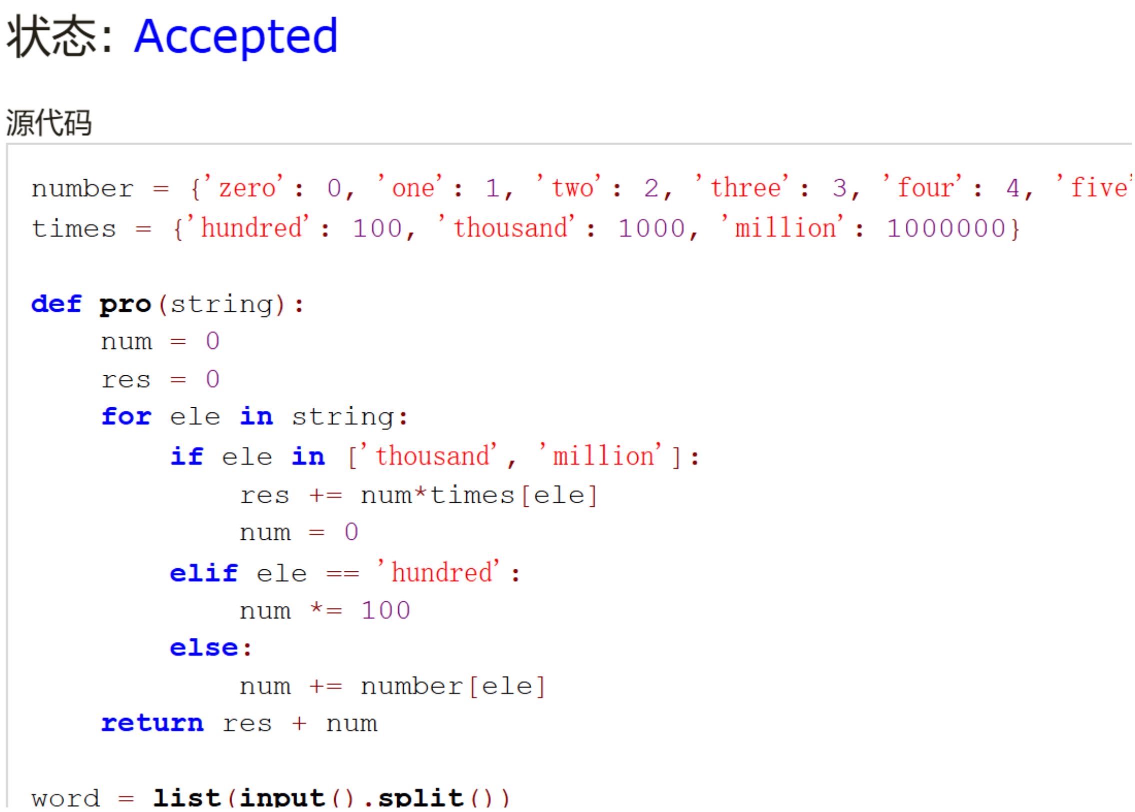
代码：

```
number = {'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, 'ninty': 90}
times = {'hundred': 100, 'thousand': 1000, 'million': 1000000}

def pro(string):
    num = 0
    res = 0
    for ele in string:
        if ele in ['thousand', 'million']:
            res += num*times[ele]
            num = 0
        elif ele == 'hundred':
            num *= 100
        else:
            num += number[ele]
    return res + num

word = list(input().split())
if word[0] == 'negative':
    del word[0]
    print(- pro(word))
else:
    print(pro(word))
```

代码运行截图 （至少包含有"Accepted"）



状态: Accepted

源代码

```
number = {'zero': 0, 'one': 1, 'two': 2, 'three': 3, 'four': 4, 'five'
times = {'hundred': 100, 'thousand': 1000, 'million': 1000000}

def pro(string):
    num = 0
    res = 0
    for ele in string:
        if ele in ['thousand', 'million']:
            res += num*times[ele]
            num = 0
        elif ele == 'hundred':
            num *= 100
        else:
            num += number[ele]
    return res + num

word = list(input().split())
```

T16528: 充实的寒假生活

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

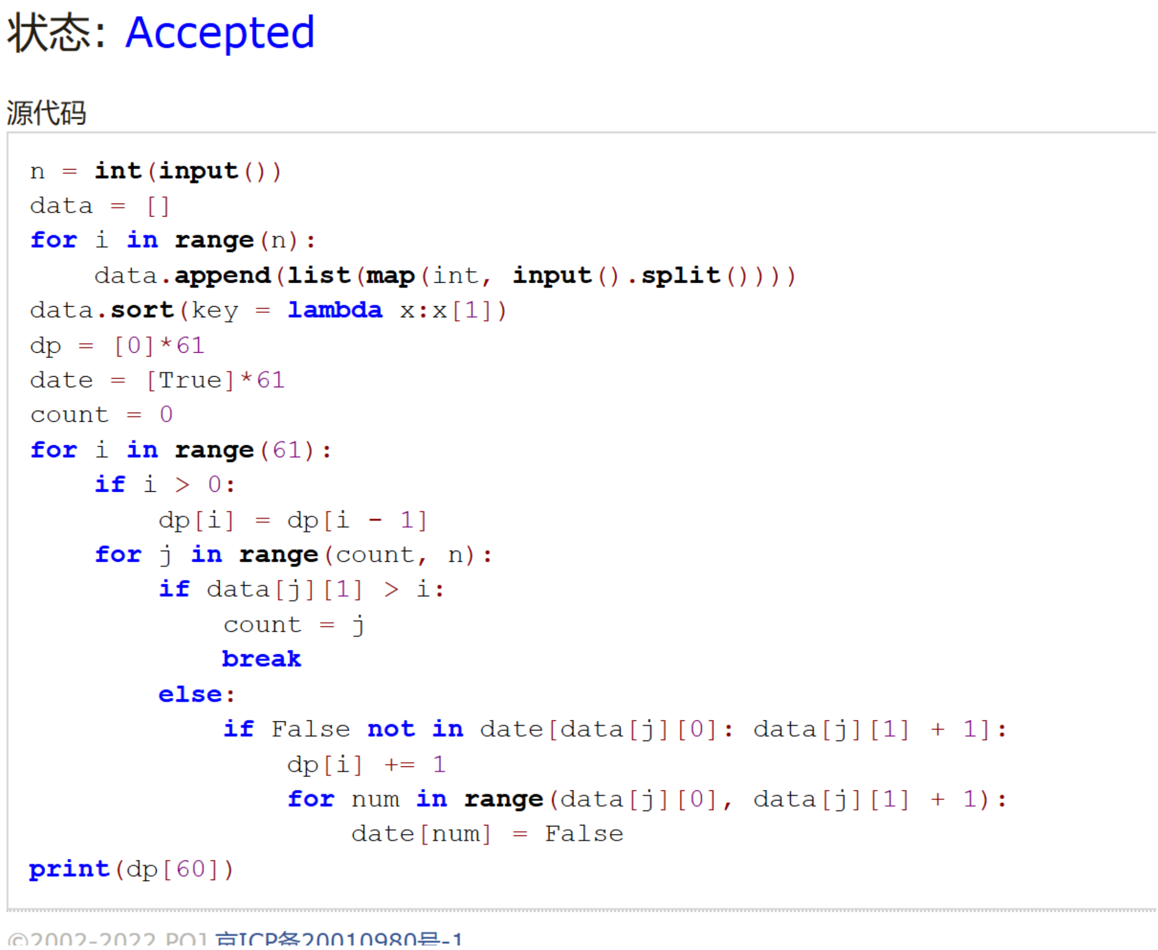
思路：

看到的时候质疑了一下是不是最难的题……非常简单，对任务根据结束天数进行排序后从第0天开始看有什么新任务可以完成就可以了

代码：

```python
n = int(input())
data = []
for i in range(n):
    data.append(list(map(int, input().split())))
data.sort(key = lambda x:x[1])
dp = [0]*61
date = [True]*61
count = 0
for i in range(61):
    if i > 0:
        dp[i] = dp[i - 1]
    for j in range(count, n):
        if data[j][1] > i:
            count = j
            break
        else:
            if False not in date[data[j][0]: data[j][1] + 1]:
                dp[i] += 1
                for num in range(data[j][0], data[j][1] + 1):
                    date[num] = False
print(dp[60])
```

代码运行截图（至少包含有"Accepted"）



状态: Accepted

源代码

```python
n = int(input())
data = []
for i in range(n):
    data.append(list(map(int, input().split())))
data.sort(key = lambda x:x[1])
dp = [0]*61
date = [True]*61
count = 0
for i in range(61):
    if i > 0:
        dp[i] = dp[i - 1]
    for j in range(count, n):
        if data[j][1] > i:
            count = j
            break
        else:
            if False not in date[data[j][0]: data[j][1] + 1]:
                dp[i] += 1
                for num in range(data[j][0], data[j][1] + 1):
                    date[num] = False
print(dp[60])
```

©2002-2022 POJ 京ICP备20010980号-1

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。

这几周没怎么做题的报应来了，明显感觉对写程序过程有些生疏了，但是dp基本思路没有忘记，虽然把6道题全部ac的时间远超两节课时间，但是最后都搞懂了。

题目难度（个人认为）: 病人排队<矩阵乘法<<翻译官<寒假生活<=零钱兑换<<<<<<打怪兽

以及讨厌字典转换题......人肉输入信息好烦