# Assignment #C: 202505114 Mock Exam

Updated 1518 GMT+8 May 14, 2025

2025 spring, Complied by <mark>同学的姓名、院系</mark>

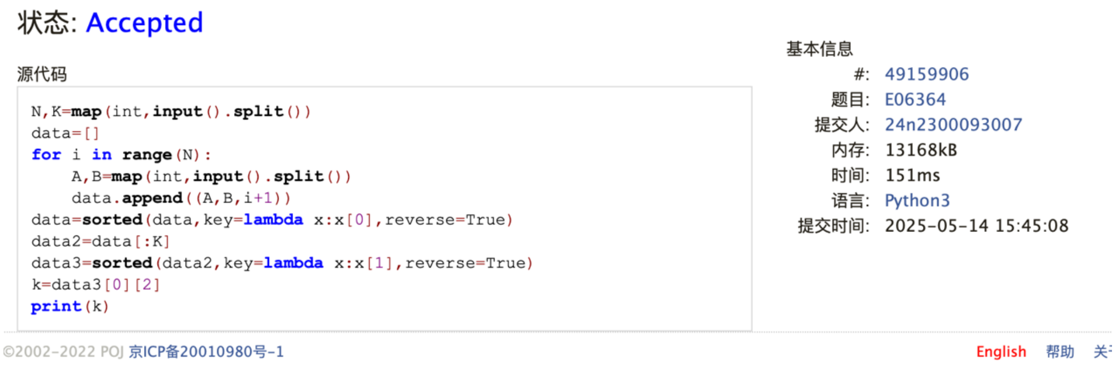
## 1. 题目

### E06364: 牛的选举

http://cs101.openjudge.cn/practice/06364/

思路:

代码:



状态: Accepted

源代码

```
N,K=map(int,input().split())
data=[]
for i in range(N):
    A,B=map(int,input().split())
    data.append((A,B,i+1))
data=sorted(data,key=lambda x:x[0],reverse=True)
data2=data[:K]
data3=sorted(data2,key=lambda x:x[1],reverse=True)
k=data3[0][2]
print(k)
```

基本信息

#: 49159906

题目: E06364

提交人: 24n2300093007

内存: 13168kB

时间: 151ms

语言: Python3

提交时间: 2025-05-14 15:45:08

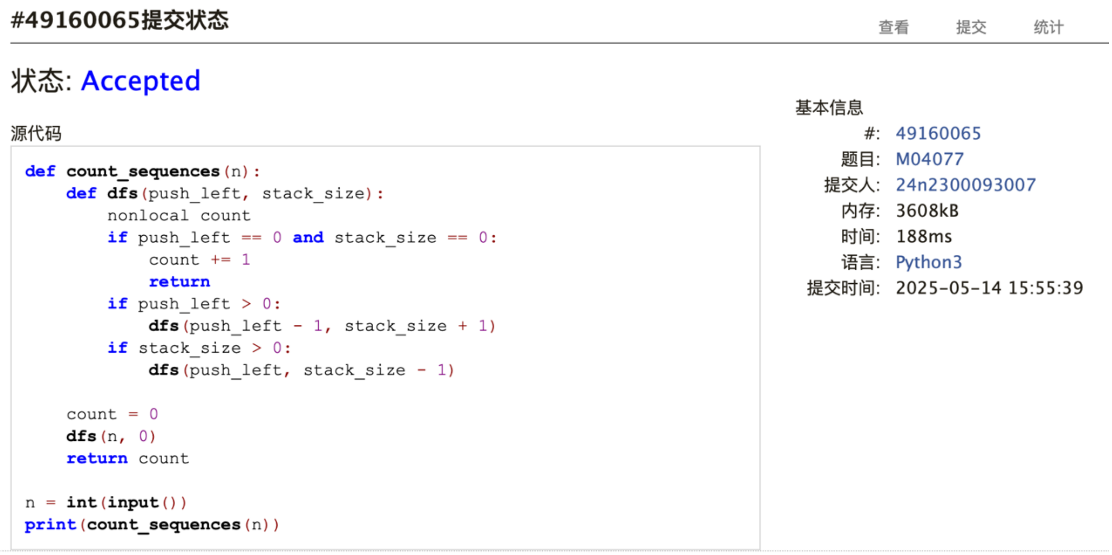
©2002-2022 POJ 京ICP备20010980号-1

### M04077: 出栈序列统计

http://cs101.openjudge.cn/practice/04077/

思路:

代码:



#49160065提交状态

状态: Accepted

源代码

```
def count_sequences(n):
    def dfs(push_left, stack_size):
        nonlocal count
        if push_left == 0 and stack_size == 0:
            count += 1
            return
        if push_left > 0:
            dfs(push_left - 1, stack_size + 1)
        if stack_size > 0:
            dfs(push_left, stack_size - 1)

    count = 0
    dfs(n, 0)
    return count

n = int(input())
print(count_sequences(n))
```

基本信息

#: 49160065

题目: M04077

提交人: 24n2300093007

内存: 3608kB

时间: 188ms

语言: Python3

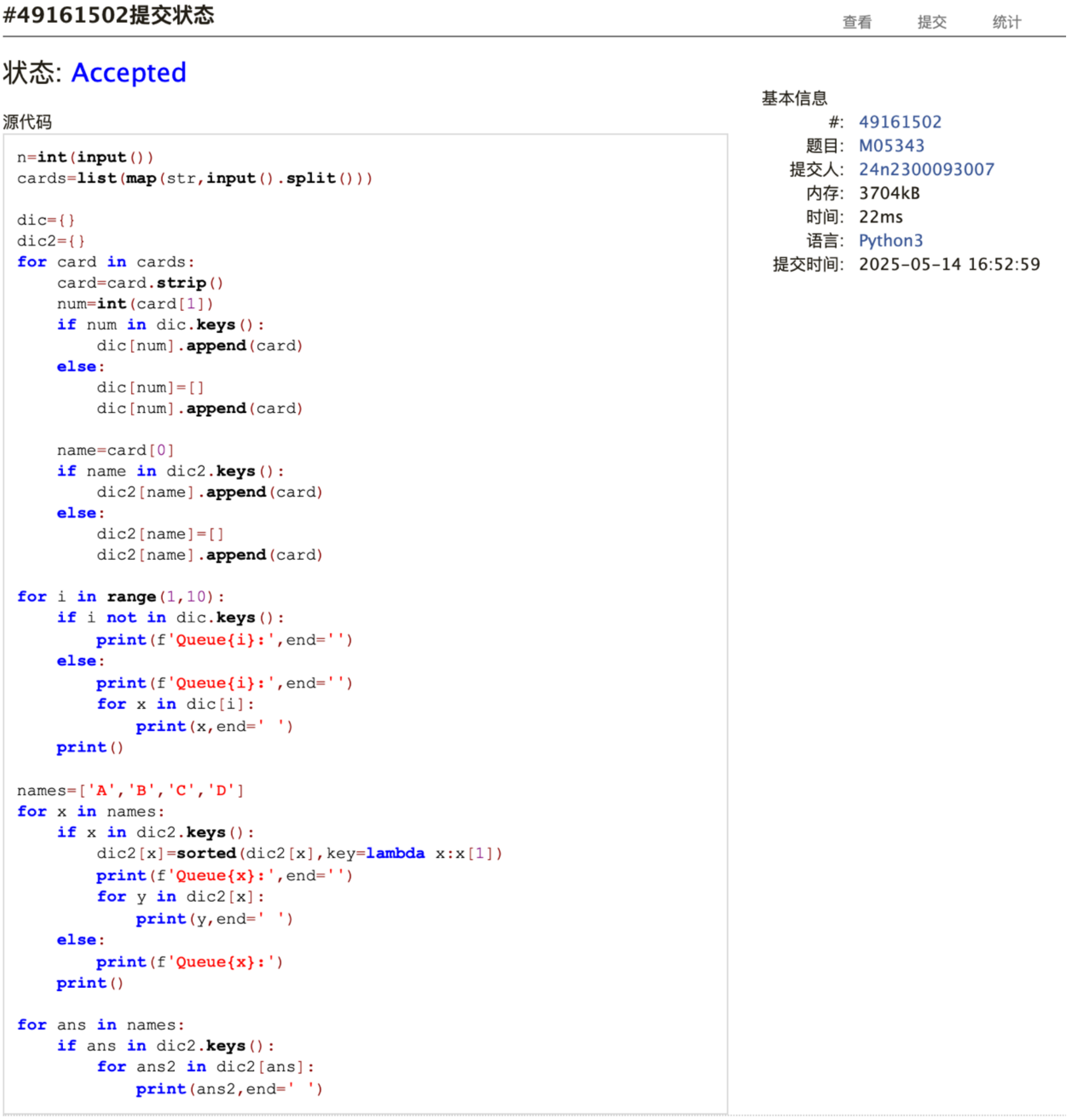
提交时间: 2025-05-14 15:55:39

### M05343:用队列对扑克牌排序

http://cs101.openjudge.cn/practice/05343/

思路:

代码:



#49161502提交状态

状态: Accepted

源代码

```
n=int(input())
cards=list(map(str,input().split()))

dic={}
dic2={}
for card in cards:
    card=card.strip()
    num=int(card[1])
    if num in dic.keys():
        dic[num].append(card)
    else:
        dic[num]=[]
        dic[num].append(card)

    name=card[0]
    if name in dic2.keys():
        dic2[name].append(card)
    else:
        dic2[name]=[]
        dic2[name].append(card)

for i in range(1,10):
    if i not in dic.keys():
        print(f'Queue{i}:',end='')
    else:
        print(f'Queue{i}:',end='')
        for x in dic[i]:
            print(x,end=' ')
    print()

names=['A','B','C','D']
for x in names:
    if x in dic2.keys():
        dic2[x]=sorted(dic2[x],key=lambda x:x[1])
        print(f'Queue{x}:',end='')
        for y in dic2[x]:
            print(y,end=' ')
    else:
        print(f'Queue{x}:')
    print()

for ans in names:
    if ans in dic2.keys():
        for ans2 in dic2[ans]:
            print(ans2,end=' ')
```

基本信息

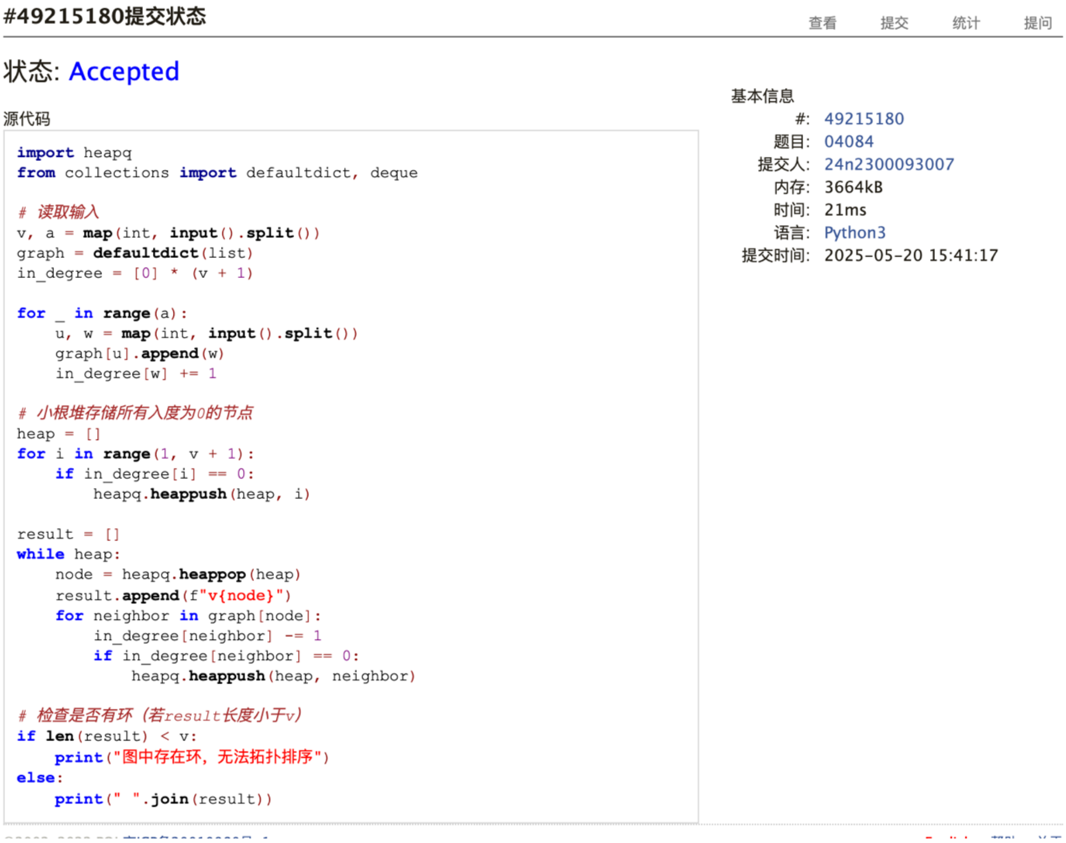
#:	49161502
题目:	M05343
提交人:	24n2300093007
内存:	3704kB
时间:	22ms
语言:	Python3
提交时间:	2025-05-14 16:52:59

### M04084: 拓扑排序

http://cs101.openjudge.cn/practice/04084/

思路:

代码:



#49215180提交状态

状态: Accepted

源代码

```python
import heapq
from collections import defaultdict, deque

# 读取输入
v, a = map(int, input().split())
graph = defaultdict(list)
in_degree = [0] * (v + 1)

for _ in range(a):
    u, w = map(int, input().split())
    graph[u].append(w)
    in_degree[w] += 1

# 小根堆存储所有入度为0的节点
heap = []
for i in range(1, v + 1):
    if in_degree[i] == 0:
        heapq.heappush(heap, i)

result = []
while heap:
    node = heapq.heappop(heap)
    result.append(f"v{node}")
    for neighbor in graph[node]:
        in_degree[neighbor] -= 1
        if in_degree[neighbor] == 0:
            heapq.heappush(heap, neighbor)

# 检查是否有环（若result长度小于v）
if len(result) < v:
    print("图中存在环，无法拓扑排序")
else:
    print(" ".join(result))
```

基本信息

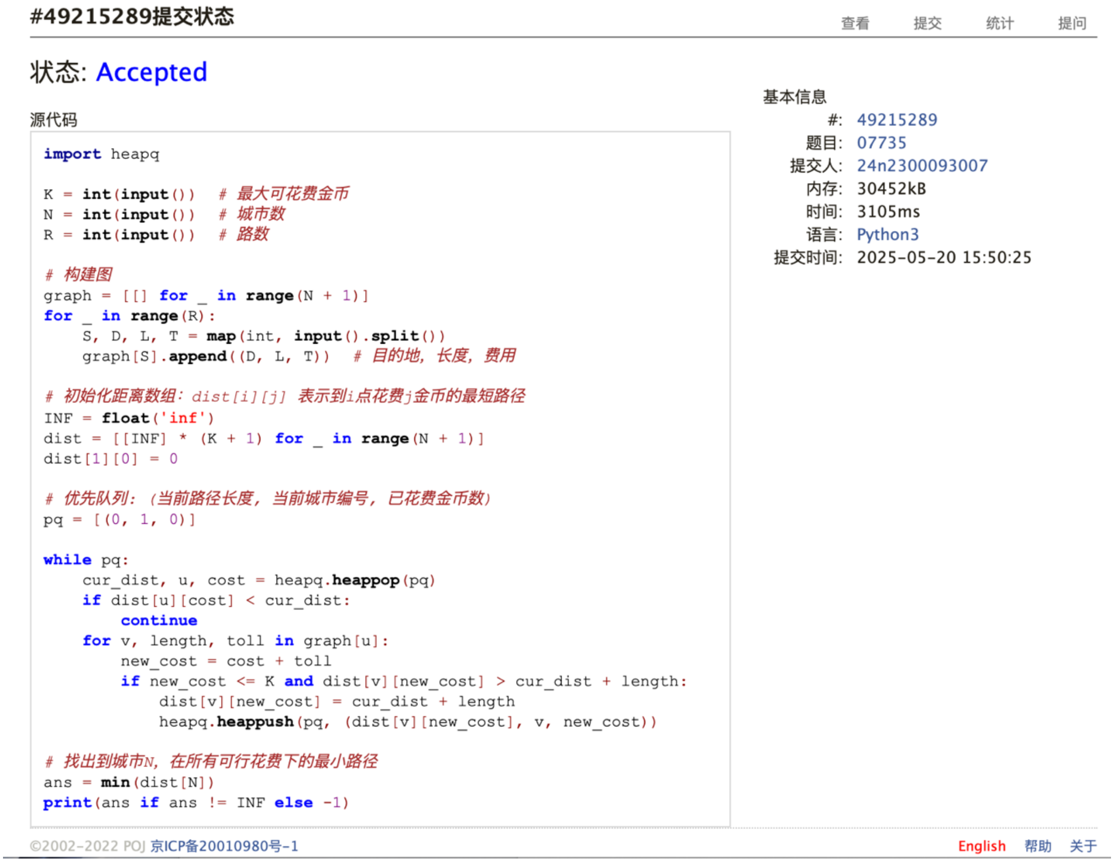
#:	49215180
题目:	04084
提交人:	24n2300093007
内存:	3664kB
时间:	21ms
语言:	Python3
提交时间:	2025-05-20 15:41:17

### M07735:道路

Dijkstra, http://cs101.openjudge.cn/practice/07735/

思路：

代码：



状态: Accepted

源代码

```
import heapq

K = int(input())  # 最大可花费金币
N = int(input())  # 城市数
R = int(input())  # 路数

# 构建图
graph = [[] for _ in range(N + 1)]
for _ in range(R):
    S, D, L, T = map(int, input().split())
    graph[S].append((D, L, T))  # 目的地, 长度, 费用

# 初始化距离数组: dist[i][j] 表示到i点花费j金币的最短路径
INF = float('inf')
dist = [[INF] * (K + 1) for _ in range(N + 1)]
dist[1][0] = 0

# 优先队列: (当前路径长度, 当前城市编号, 已花费金币数)
pq = [(0, 1, 0)]

while pq:
    cur_dist, u, cost = heapq.heappop(pq)
    if dist[u][cost] < cur_dist:
        continue
    for v, length, toll in graph[u]:
        new_cost = cost + toll
        if new_cost <= K and dist[v][new_cost] > cur_dist + length:
            dist[v][new_cost] = cur_dist + length
            heapq.heappush(pq, (dist[v][new_cost], v, new_cost))

# 找出到城市N, 在所有可行花费下的最小路径
ans = min(dist[N])
print(ans if ans != INF else -1)
```

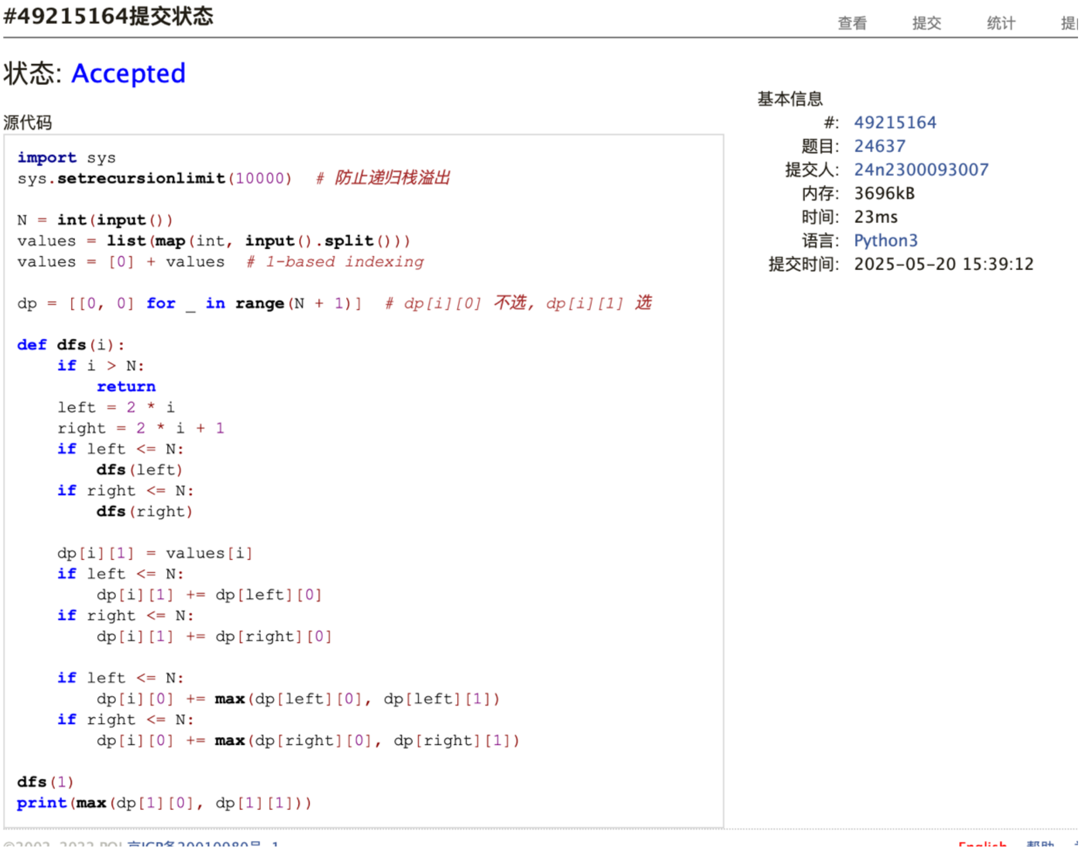
基本信息
#: 49215289
题目: 07735
提交人: 24n2300093007
内存: 30452kB
时间: 3105ms
语言: Python3
提交时间: 2025-05-20 15:50:25

### T24637:宝藏二叉树

dp, http://cs101.openjudge.cn/practice/24637/

思路:

代码:



#49215164提交状态

状态: Accepted

源代码

```
import sys
sys.setrecursionlimit(10000)  # 防止递归栈溢出

N = int(input())
values = list(map(int, input().split()))
values = [0] + values  # 1-based indexing

dp = [[0, 0] for _ in range(N + 1)]  # dp[i][0] 不选, dp[i][1] 选

def dfs(i):
    if i > N:
        return
    left = 2 * i
    right = 2 * i + 1
    if left <= N:
        dfs(left)
    if right <= N:
        dfs(right)

    dp[i][1] = values[i]
    if left <= N:
        dp[i][1] += dp[left][0]
    if right <= N:
        dp[i][1] += dp[right][0]

    if left <= N:
        dp[i][0] += max(dp[left][0], dp[left][1])
    if right <= N:
        dp[i][0] += max(dp[right][0], dp[right][1])

dfs(1)
print(max(dp[1][0], dp[1][1]))
```

基本信息

#:	49215164
题目:	24637
提交人:	24n2300093007
内存:	3696kB
时间:	23ms
语言:	Python3
提交时间:	2025-05-20 15:39:12

## 2．学习总结和收获

这次月考只有 AC2，感觉二叉树、defaultdict 需要多多复习。机考前我一定会复习完相关结构。