

# Assignment #D: Mock Exam

Updated 1955 GMT+8 Dec 5, 2025

2025 fall, Compiled by 魏佳亮 2400011474

## 说明:

1. Dec月考: AC3 (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. 解题与记录: 对于每一个题目, 请提供其解题思路(可选), 并附上使用Python或C++编写的源代码(确保已在OpenJudge, Codeforces, LeetCode等平台上获得Accepted)。请将这些信息连同显示“Accepted”的截图一起填写到下方的作业模板中。(推荐使用Typora <https://typoraio.cn> 进行编辑, 当然你也可以选择Word。)无论题目是否已通过, 请标明每个题目大致花费的时间。
3. 提交安排: 提交时, 请首先上传PDF格式的文件, 并将.md或.doc格式的文件作为附件上传至右侧的“作业评论”区。确保你的Canvas账户有一个清晰可见的本人头像, 提交的文件为PDF格式, 并且“作业评论”区包含上传的.md或.doc附件。
4. 延迟提交: 如果你预计无法在截止日期前提交作业, 请提前告知具体原因。这有助于我们了解情况并可能为你提供适当的延期或其他帮助。

请按照上述指导认真准备和提交作业, 以保证顺利完成课程要求。

## 1. 题目

### E02734:十进制到八进制

<http://cs101.openjudge.cn/practice/02734>

思路: 直接除呗

代码

```
from math import log2
a=int(input())
n=int(log2(a)//3)
l=[]
for i in range(1,n+1)[::-1]:
    l.append(a//(8**i))
    a=a%(8**i)
l.append(a)
print(''.join([str(x) for x in l]))
```

代码运行截图 (至少包含有"Accepted")

#51148009提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```
from math import log2
a=int(input())
n=int(log2(a)//3)
l=[]
for i in range(1,n+1)[::-1]:
    l.append(a//(8**i))
    a=a%(8**i)
l.append(a)
print(''.join([str(x) for x in l]))
```

基本信息

#: 51148009  
题目: E02734  
提交人: 24n2400011474  
内存: 3640kB  
时间: 23ms  
语言: Python3  
提交时间: 2025-12-05 17:31:44

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M21509:序列的中位数

heap, <http://cs101.openjudge.cn/practice/21509>

思路: 维护左右两个堆

代码

```
import heapq
n=int(input())
l,r=[],[]
heapq.heapify(l)
heapq.heapify(r)
s=list(map(int,input().split()))
a,b=0,0
for i in range(n):
    if i==0:
        heapq.heappush(l,-s[0])
        a+=1
    if a==b and i:
        y=heapq.heappop(r)
        if s[i]<=y:
            heapq.heappush(r,y)
            heapq.heappush(l,-s[i])
        else:
            heapq.heappush(r,s[i])
            heapq.heappush(l,-y)
        a+=1
    elif i:
        x=heapq.heappop(l)
        if s[i]>=-x:
            heapq.heappush(l,x)
```

```

        heapq.heappush(r,s[i])
        b+=1
    else:
        heapq.heappush(l,-s[i])
        heapq.heappush(r,-x)
        b+=1
if i%2==0:
    x=heapq.heappop(l)
    print(-x)
    heapq.heappush(l,x)

```

代码运行截图 (至少包含有"Accepted")

#51147832提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: Accepted

源代码

```

import heapq
n=int(input())
l,r=[],[]
heapq.heapify(l)
heapq.heapify(r)
s=list(map(int,input().split()))
a,b=0,0
for i in range(n):
    if i==0:
        heapq.heappush(l,-s[0])
        a+=1
    if a==b and i:
        y=heapq.heappop(r)
        if s[i]<=y:
            heapq.heappush(r,y)
            heapq.heappush(l,-s[i])
        else:
            heapq.heappush(r,s[i])
            heapq.heappush(l,-y)
        a+=1
    elif i:
        x=heapq.heappop(l)
        if s[i]>=-x:
            heapq.heappush(l,x)
            heapq.heappush(r,s[i])
            b+=1
        else:
            heapq.heappush(l,-s[i])
            heapq.heappush(r,-x)
            b+=1
    if i%2==0:
        x=heapq.heappop(l)
        print(-x)
        heapq.heappush(l,x)

```

基本信息

#: 51147832  
 题目: M21509  
 提交人: 24n2400011474  
 内存: 15600kB  
 时间: 218ms  
 语言: Python3  
 提交时间: 2025-12-05 17:26:15

©2002-2022 POJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## M27306: 植物观察

disjoint set, bfs, <http://cs101.openjudge.cn/practice/27306/>

思路: 维护同源/不同源两个并查集

代码

```
n,m=map(int,input().split())
l=[i for i in range(2*n)]
def find(x):
    if l[x]!=x:
        l[x]=find(l[x])
    return l[x]
def union(x,y):
    a,b=find(x),find(y)
    l[a]=b
    l[a+n]=b+n
def split(x,y):
    a,b=find(x),find(y)
    l[a+n]=b
    l[b+n]=a
k=True
for _ in range(m):
    x,y,z=map(int,input().split())
    if not z:
        if find(l[x+n])==find(l[y]):
            print('NO')
            k=False
            break
        union(x,y)
    else:
        if find(l[x])==find(l[y]):
            print('NO')
            k=False
            break
        split(x,y)
if k:
    print('YES')
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

基本信息

#: 51148422

题目: M27306

提交人: 24n2400011474

内存: 3700kB

时间: 33ms

语言: Python3

提交时间: 2025-12-05 17:49:06

源代码

```
n,m=map(int,input().split())
l=[i for i in range(2*n)]
def find(x):
    if l[x]!=x:
        l[x]=find(l[x])
    return l[x]
def union(x,y):
    a,b=find(x),find(y)
    l[a]=b
    l[a+n]=b+n
def split(x,y):
    a,b=find(x),find(y)
    l[a+n]=b
    l[b+n]=a
k=True
for _ in range(m):
    x,y,z=map(int,input().split())
    if not z:
        if find(l[x+n])==find(l[y]):
            print('NO')
            k=False
            break
        union(x,y)
    else:
        if find(l[x])==find(l[y]):
            print('NO')
            k=False
            break
        split(x,y)
if k:
    print('YES')
```

## M29740:神经网络

Topological order, <http://cs101.openjudge.cn/practice/29740/>思路：卡了半天啊。。。实际上就是 $c=\max(c,0)$ 

代码

```
n,p=map(int,input().split())
c=[0 for _ in range(n)]
u=[0 for _ in range(n)]
for i in range(n):
    a,b=map(int,input().split())
    c[i]=a
    u[i]=b
a=[0 for _ in range(n)]
b={i:[] for i in range(n)}
l=[[0]*n for _ in range(n)]
k=[[0]*n for _ in range(n)]
for _ in range(p):
```

```

i,j,w=map(int,input().split())
l[i-1][j-1]+=w
k[i-1][j-1]=1
for i in range(n):
    for j in range(n):
        if k[i][j]:
            a[j]+=1
            b[i].append((j,l[i][j]))
s=[]
for i in range(n):
    if a[i]==0:
        s.append(i)
    c[i]=max(0,c[i])
ans=[]
def solve(x):
    if not b[x] and c[x]>0:
        ans.append((x,c[x]))
        return
    for y in b[x]:
        a[y[0]]-=1
        c[y[0]]+=c[x]*y[1]
        if not a[y[0]]:
            c[y[0]]=max(0,c[y[0]]-u[y[0]])
            solve(y[0])
for x in s:
    solve(x)
if not ans or a!=[0]*n:
    print('NULL')
else:
    k=sorted(ans)
    for x,y in k:
        print(x+1,y)

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n,p=map(int,input().split())
c=[0 for _ in range(n)]
u=[0 for _ in range(n)]
for i in range(n):
    a,b=map(int,input().split())
    c[i]=a
    u[i]=b
a=[0 for _ in range(n)]
b={i:[] for i in range(n)}
l=[[0]*n for _ in range(n)]
k=[[0]*n for _ in range(n)]
for _ in range(p):
    i,j,w=map(int,input().split())
    l[i-1][j-1]+=w
    k[i-1][j-1]=1
for i in range(n):
    for j in range(n):
        if k[i][j]:
            a[j]+=1
            b[i].append((j,l[i][j]))
s=[]
for i in range(n):
    if a[i]==0:
        s.append(i)
        c[i]=max(0,c[i])
ans=[]
def solve(x):
    if not b[x] and c[x]>0:
        ans.append((x,c[x]))
        return
    for y in b[x]:
        a[y[0]]-=1
        c[y[0]]+=c[x]*y[1]
        if not a[y[0]]:
            c[y[0]]=max(0,c[y[0]]-u[y[0]])
            solve(y[0])
for x in s:
```

基本信息

#: 51202308

题目: 29740

提交人: 24n2400011474

内存: 3932kB

时间: 32ms

语言: Python3

提交时间: 2025-12-09 10:42:11

## T27351:01最小生成树

mst, <http://cs101.openjudge.cn/practice/27351/>

思路: dfs, 除了时间长点都很好

代码

```
from collections import deque
n,m=map(int,input().split())
s=[set() for _ in range(n)]
for _ in range(m):
    x,y=map(int,input().split())
    s[x-1].add(y-1)
    s[y-1].add(x-1)
l=set(range(n))
ans=-1
```

```

while l:
    ans+=1
    x=l.pop()
    q=deque([x])
    while q:
        y=q.popleft()
        t=l-s[y]
        for z in list(t):
            l.remove(z)
            q.append(z)
print(ans)

```

代码运行截图 (至少包含有"Accepted")

#51200506提交状态

[查看](#) [提交](#) [统计](#) [提问](#)

状态: **Accepted**

源代码

```

from collections import deque
n,m=map(int,input().split())
s=[set() for _ in range(n)]
for _ in range(m):
    x,y=map(int,input().split())
    s[x-1].add(y-1)
    s[y-1].add(x-1)
l=set(range(n))
ans=-1
while l:
    ans+=1
    x=l.pop()
    q=deque([x])
    while q:
        y=q.popleft()
        t=l-s[y]
        for z in list(t):
            l.remove(z)
            q.append(z)
print(ans)

```

基本信息

#: 51200506  
 题目: 27351  
 提交人: 24n2400011474  
 内存: 61716kB  
 时间: 4753ms  
 语言: Python3  
 提交时间: 2025-12-08 23:12:19

©2002-2022 PQJ 京ICP备20010980号-1

[English](#) [帮助](#) [关于](#)

## T30193:哈密顿激活层

DFS+剪枝, <http://cs101.openjudge.cn/practice/30193/>

思路: 剪枝思路就是不能提前占坑+不能和未来要到的点距离太远 (超过时间差)

代码

```

n,m,k,b=map(int,input().split())
l=[]
w=[]
s=[[0]*m for _ in range(n)]

```



```

for _ in range(k):
    r,c,t=map(int,input().split())
    if t==1:
        x,y=r-1,c-1
    l.append((r-1,c-1,t))
for _ in range(b):
    r,c=map(int,input().split())
    s[r-1][c-1]=-1
r=[(0,-1),(0,1),(-1,0),(1,0)]
s[x][y]=1
def solve(s,x,y,t):
    if t==n*m-b:
        return s
    for dx,dy in r:
        if 0<=x+dx<n and 0<=y+dy<m and not s[x+dx][y+dy]:
            p=True
            for i,j,k in l:
                if (k>=t+1 and abs(x+dx-i)+abs(y+dy-j)>k-t-1) or (x+dx==i and y+dy==j and
k!=t+1):
                    p=False
                    break
            if p:
                s[x+dx][y+dy]=t+1
                q=solve(s,x+dx,y+dy,t+1)
                if q:
                    return q
                s[x+dx][y+dy]=0
    return False
q=solve(s,x,y,1)
if q:
    a=[0]*(n*m-b)
    for i in range(n):
        for j in range(m):
            if q[i][j]!=-1:
                a[q[i][j]-1]=(i+1,j+1)
    for x,y in a:
        print(x,y)
else:
    print(-1)

```

代码运行截图 (至少包含有"Accepted")

状态: **Accepted**

源代码

```
n,m,k,b=map(int,input().split())
l=[]
w=[]
s=[[0]*m for _ in range(n)]
for _ in range(k):
    r,c,t=map(int,input().split())
    if t==1:
        x,y=r-1,c-1
        l.append((r-1,c-1,t))
for _ in range(b):
    r,c=map(int,input().split())
    s[r-1][c-1]=-1
r=[(0,-1),(0,1),(-1,0),(1,0)]
s[x][y]=1
def solve(s,x,y,t):
    if t==n*m-b:
        return s
    for dx,dy in r:
        if 0<=x+dx<n and 0<=y+dy<m and not s[x+dx][y+dy]:
            p=True
            for i,j,k in l:
                if (k>=t+1 and abs(x+dx-i)+abs(y+dy-j)>k-t-1) or (x+dx==i and y+dy==j and k==t):
                    p=False
                    break
            if p:
                s[x+dx][y+dy]=t+1
                q=solve(s,x+dx,y+dy,t+1)
                if q:
                    return q
                s[x+dx][y+dy]=0
    return False
q=solve(s,x,y,1)
if q:
    a=[0]*(n*m-b)
    for i in range(n):
        for j in range(m):
            if q[i][j]!=-1:
                a[q[i][j]-1]=(i+1,j+1)
```

基本信息

#: [51203031](#)  
题目: [30193](#)  
提交人: [24n2400011474](#)  
内存: 3756kB  
时间: 7579ms  
语言: [Python3](#)  
提交时间: 2025-12-09 11:47:28

## 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2025fall每日选做”、CF、LeetCode、洛谷等网站题目。

一整个破防了，才ac3，开始最小生成树还以为就是dijkstra，然后一直runtime error，忘了有递归这事了，但是主要是 $n^2$ 的union确实不行。后面三个题研究了得一共6个小时。。。最后发现其实不怎么难，时间也不卡的不太严。。。希望期末考试能发挥好点（最近每日选做没咋做因为之前都做过了，然后我最近也很忙就搁置了）