

# assignmentD

## Assignment #D : Mock Exam

Updated 1955 GMT+8 Dec 5, 2025

2025 fall, Compiled by 杨知进 物理学院

### 说明：

1. Dec月考：AC0（工作会议，未参加）。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：

代码

```
a = int(input().strip())
print(oct(a)[2:])
```

代码运行截图（至少包含有"Accepted"）

## #51216233提交状态

---

状态: Accepted

源代码

```
a = int(input().strip())
print(oct(a)[2:])
```

*image.png*

用时: 5min

## M21509:序列的中位数

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

思路:

代码

```
import heapq

N = int(input())
A = list(map(int, input().split()))

max_heap = []
min_heap = []
for i in range(N):
    num = A[i]
    if not max_heap or num <= -max_heap[0]:
        heapq.heappush(max_heap, -num)
    else:
        heapq.heappush(min_heap, num)

    if len(max_heap) > len(min_heap) + 1:
        heapq.heappush(min_heap, -heapq.heappop(max_heap))
    elif len(max_heap) < len(min_heap):
        heapq.heappush(max_heap, -heapq.heappop(min_heap))
    if i % 2 == 0:
        print(-max_heap[0])
```

代码运行截图 (至少包含有"Accepted")

# #51216261提交状态

---

状态: Accepted

*image-1.png*

用时: 20min左右

## M27306: 植物观察

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

思路:

代码

```
from collections import deque

def solve():
    n, m = map(int, input().split())
    graph = [[] for _ in range(n)]
    for _ in range(m):
        u, v, w = map(int, input().split())
        graph[u].append((v, w))
        graph[v].append((u, w))
    color = [-1] * n
    for i in range(n):
        if color[i] == -1:
            q = deque([i])
            color[i] = 0
            while q:
                node = q.popleft()
                for nei, rel in graph[node]:
                    if color[nei] == -1:
                        if rel == 0:
                            color[nei] = color[node]
                        else:
                            color[nei] = 1 - color[node]
                        q.append(nei)
                    else:
                        if rel == 0 and color[nei] != color[node]:
                            print("NO")
                            return
```

```
        if rel == 1 and color[nei] == color[node]:
            print("NO")
            return

    print("YES")

solve()
```

代码运行截图 (至少包含有"Accepted")

## #51216306提交状态

---

状态: Accepted

*image-2.png*

用时: 20min左右

### M29740:神经网络

Topological order, <http://cs101.openjudge.cn/practice/29740/>

思路:

代码

代码运行截图 (至少包含有"Accepted")

### T27351:01最小生成树

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

思路:

代码

```
from sys import stdin, setrecursionlimit
from collections import deque
import sys

setrecursionlimit(10**6)
input = lambda: stdin.readline().strip()
```

```

def solve():
    n, m = map(int, input().split())
    edges = set()
    for _ in range(m):
        a, b = map(int, input().split())
        a -= 1
        b -= 1
        if a > b:
            a, b = b, a
        edges.add((a, b))

    unvisited = set(range(n))
    t = 0

    while unvisited:
        t += 1
        start = unvisited.pop()
        q = deque([start])
        while q:
            u = q.popleft()
            to_remove = []
            for v in unvisited:
                a, b = (u, v) if u < v else (v, u)
                if (a, b) not in edges:
                    to_remove.append(v)
            for v in to_remove:
                unvisited.remove(v)
                q.append(v)
        print(t - 1)

solve()

```

代码运行截图 (至少包含有"Accepted")

## #51216342提交状态

---

状态: Accepted

*image-3.png*

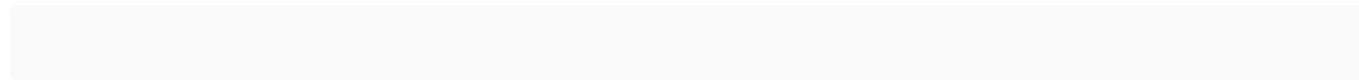
用时: 30min左右

**T30193:哈密顿激活层**

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

思路:

代码



代码运行截图 (至少包含有"Accepted")

## 2. 学习总结和收获

还在跟各种细节搏斗.....