

# Assignment #7: 20250402 Mock Exam

Updated 1624 GMT+8 Apr 2, 2025

2025 spring, Compiled by 同学的姓名、院系

## 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. **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

### E05344:最后的最后

<http://cs101.openjudge.cn/practice/05344/>

思路:

代码:

```
n,k=[int(x) for x in input().split()]
cnt=k
lst=[int(i) for i in range(1,n+1)]
ans=[]
while len(lst)>1:
    if k==1:
        ans.append(lst.pop(0))
        k=cnt
    lst.append(lst.pop(0))
    k-=1
print(*ans)
```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```
n,k=[int(x) for x in input().split()]
cnt=k
lst=[int(i) for i in range(1,n+1)]
ans=[]
while len(lst)>1:
    if k==1:
        ans.append(lst.pop(0))
        k=cnt
    lst.append(lst.pop(0))
    k-=1
print(*ans)
```

©2002-2022 POJ 京ICP备20010980号-1

## M02774: 木材加工

binary search, <http://cs101.openjudge.cn/practice/02774/>

思路:

代码:

```
n,k=[int(x) for x in input().split()]
lst=[]
for i in range(n):
    lst.append(int(input()))
lst.sort()
record=lst[-1]
cnt=0
for i in lst:
    cnt+=i//record

while cnt<k:
    cnt=0
    record-=1
    if record==0:
```

```

        break
    for i in lst:
        cnt+=i//record

print(record)

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

n,k=[int(x) for x in input().split()]
lst=[]
for i in range(n):
    lst.append(int(input()))
lst.sort()
record=lst[-1]
cnt=0
for i in lst:
    cnt+=i//record

while cnt<k:
    cnt=0
    record-=1
    if record==0:
        break
    for i in lst:
        cnt+=i//record

```

## M07161:森林的带度数层次序列存储

tree, <http://cs101.openjudge.cn/practice/07161/>

思路:

代码:

```

class TreeNode:
    def __init__(self, val=0, degree=0, children=[]):
        self.val=val
        self.degree=degree
        self.children=children

```

```

def build(s):
    tree=[None]*(len(s)//2)
    for i in range(len(tree)):
        tree[i]=TreeNode(s[2*i],s[2*i+1],[])
    record=tree[1:]
    for i in range(len(tree)):
        for j in range(int(tree[i].degree)):
            tree[i].children.append(record.pop(0))
    return tree[0]

def f(root):
    ans=[]
    if root.children:
        for i in root.children:
            ans+=f(i)
    ans+=root.val
    return ans

n=int(input())
out=[]
for _ in range(n):
    s=[i for i in input().split()]
    root=build(s)
    out+=f(root)
print(*out)

```

代码运行截图 (至少包含有"Accepted")

状态: Accepted

源代码

```

class TreeNode:
    def __init__(self, val=0, degree=0, children=[]):
        self.val=val
        self.degree=degree
        self.children=children

def build(s):
    tree=[None]*(len(s)//2)
    for i in range(len(tree)):
        tree[i]=TreeNode(s[2*i],s[2*i+1],[])
    record=tree[1:]
    for i in range(len(tree)):
        for j in range(int(tree[i].degree)):
            tree[i].children.append(record.pop(0))
    return tree[0]

def f(root):
    ans=[]
    if root.children:
        for i in root.children:
            ans+=f(i)
    ans+=root.val
    return ans

```

## M18156:寻找离目标数最近的两数之和

two pointers, <http://cs101.openjudge.cn/practice/18156/>

思路:

代码:

```
t=int(input())
s=[int(x) for x in input().split()]
s.sort()
ans=[(float('inf'),0)]
l,r=0,len(s)-1
while l<r:
    record=s[l]+s[r]
    if abs(t-record)<ans[0][0]:
        ans.pop(0)
        ans.append([abs(t-record),record])
    elif abs(t-record)==ans[0][0]:
        ans.append([abs(t-record),record])
    if record<=t:
        l+=1
    else:
        r-=1
ans.sort()
print(ans[0][1])
```

代码运行截图 (至少包含有"Accepted")



```

r1st=prime(10001)
for _ in range(t):
    n=int(input())
    ans=[]
    st=n
    for cnt in range(2,n):
        if str(cnt)[-1]=='1':
            st=cnt
            break
    l1st=[int(i) for i in range(st,n,10)]
    for i in l1st:
        if r1st[i]:
            ans.append(i)
    print(f'Case{_{+1}}:')
    if ans:
        print(' '.join(map(str,ans)))
    else:
        print('NULL')

```

代码运行截图 (至少包含有"Accepted")

#48799981提交记录

状态: Accepted

源代码

```

def prime(n):
    lst=[True]*(n+1)
    primes=[]
    lst[1]=False
    for i in range(2,n+1):
        if lst[i]:
            primes.append(i)
            for p in primes:
                if i*p>n:
                    break
                lst[i*p]=False
                if i%p==0:
                    break
    return lst

t=int(input())
r1st=prime(10001)
for _ in range(t):
    n=int(input())
    ans=[]
    st=n

```

## M28127:北大夺冠

hash table, <http://cs101.openjudge.cn/practice/28127/>

思路:

代码:

```
m=int(input())
dic={}
for _ in range(m):
    team,q,res=[i for i in input().split(',')]
    if team not in dic:
        if res=='yes':
            dic[team]=[1,1,q] ##做对题目数, 提交次数, 正确题目
        else:
            dic[team]=[0,1,'']
    else:
        if res=='yes':
            if q in dic[team][2]:
                dic[team][1]+=1
            else:
                dic[team][0]+=1
                dic[team][1]+=1
                dic[team][2]+=q
        else:
            dic[team][1]+=1
key,value=list(dic.keys()),list(dic.values())
lst=[]
for i in range(len(key)):
    lst.append([value[i][0],value[i][1],key[i]])
lst.sort(key=lambda x:(-x[0],x[1],x[2]))
for i in range(min(12,len(key))):
    print(i+1,lst[i][2],lst[i][0],lst[i][1])
```

代码运行截图 == (AC代码截图, 至少包含有"Accepted") ==



---

状态: Accepted

源代码

```
m=int(input())
dic={}
for _ in range(m):
    team,q,res=[i for i in input().split(',') ]
    if team not in dic:
        if res=='yes':
            dic[team]=[1,1,q]  ##做对题目数, 提交次数, 正确题目
        else:
            dic[team]=[0,1,'']
    else:
        if res=='yes':
            if q in dic[team][2]:
                dic[team][1]+=1
            else:
                dic[team][0]+=1
                dic[team][1]+=1
                dic[team][2]+=q
        else:
            dic[team][1]+=1
key,value=list(dic.keys()),list(dic.values())
lst=[]
for i in range(len(key)):
    lst.append([value[i][0],value[i][1],key[i]])
    _ _ _ _ _
```

## 2. 学习总结和收获

如果发现作业题目相对简单, 有否寻找额外的练习题目, 如“数算2025spring每日选做”、LeetCode、Codeforces、洛谷等网站上的题目。

这次月考第二题和第五题都因为非常蠢的bug卡了好久好久, 导致考试时根本没看最后一道北大夺冠, 课下写时发现这题相当简单, 只不过是麻烦了一点, 哎

树那题还是不熟练, 回来后重新看了看讲义才写出来