

# assignment9

## Assignment #9: Mock Exam立冬

Updated 1856 GMT+8 Nov 7, 2025

2025 fall, Complied by 杨知进 物理学院

### 说明:

1. Nov月考: AC0 (部门工作会议, 没有去考试 😢)。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

### M02255: 重建二叉树

<http://cs101.openjudge.cn/practice/02255/>

思路:

主要是细致, 思路不乱

代码

```
import sys
def build_tree(preorder, inorder):
    if not preorder:
```

```

        return ""
root = preorder[0]
root_index = inorder.index(root)
left_inorder = inorder[:root_index]
right_inorder = inorder[root_index + 1:]
left_preorder = preorder[1:1 + len(left_inorder)]
right_preorder = preorder[1 + len(left_inorder):]
left_postorder = build_tree(left_preorder, left_inorder)
right_postorder = build_tree(right_preorder, right_inorder)
return left_postorder + right_postorder + root

for line in sys.stdin:
    line = line.strip()
    if not line:
        continue
    preorder, inorder = line.split()
    postorder = build_tree(preorder, inorder)
    print(postorder)

```

代码运行截图 (至少包含有"Accepted")

**#50800925提交状态**

---

状态: Accepted

*image.png*

用时: 20min左右

**M02774: 木材加工**

<http://cs101.openjudge.cn/practice/02774/>

思路:

二分查找

代码

```

n,k=map(int,input().split())
logs = []
for i in range(n):
    logs.append(int(input()))
if sum(logs) < k:

```

```

print(0)
else:
    left, right = 1, max(logs)
    out = 0
    while left <= right:
        mid = (left + right) // 2
        total = sum(log // mid for log in logs)
        if total >= k:
            out = mid
            left = mid + 1
        else:
            right = mid - 1
print(out)

```

代码运行截图 (至少包含有"Accepted")

**#50800996提交状态**

---

状态: Accepted

*image-1.png*

用时: 13min左右

**M02788: 二叉树 (2)**

<http://cs101.openjudge.cn/practice/02788/>

思路:

难点在于读懂题目要计算的东西，找到算法。一层一层算并转移到下一层

代码

```

import sys

def f(m, n):
    if m > n:
        return 0
    count = 0
    left = m
    right = m

    while left <= n:

```

```
        count += min(right, n) - left + 1
        left = left * 2
        right = right * 2 + 1
    return count

for line in sys.stdin:
    m, n = map(int, line.split())
    if m == 0 and n == 0:
        break
    out = f(m, n)
    print(out)
```

代码运行截图 (至少包含有"Accepted")

## #50801114提交状态

状态: Accepted

*image-2.png*

用时: 25min左右

## M04081: 树的转换

<http://cs101.openjudge.cn/practice/04081/>

思路:

代码

代码运行截图 (至少包含有"Accepted")

## M04117: 简单的整数划分问题

dfs, dp, <http://cs101.openjudge.cn/practice/04117/>

思路:

并不是很简单.....自己做陷进去总是意识不到这是一个动态规划/dp问题。借助ai才开朗了，把它想得太简单去找一个一维的递推去了，实际上还得是二维。

代码

```
import sys

def f(n):
    dp = [[0] * (n + 1) for _ in range(n + 1)]
    for i in range(n + 1):
        for j in range(n + 1):
            if i == 0:
                dp[i][j] = 1
            elif j == 0:
                dp[i][j] = 0
            elif j > i:
                dp[i][j] = dp[i][i]
            else:
                dp[i][j] = dp[i][j - 1] + dp[i - j][j]
    return dp[n][n]
for line in sys.stdin:
    n = int(line.strip())
    print(f(n))
```

代码运行截图 (至少包含有"Accepted")

## #50801420提交状态

---

状态: Accepted

*image-5.png*

用时: 37min左右

## M04137:最小新整数

monotonous-stack, <http://cs101.openjudge.cn/practice/04137/>

思路:

stack, 从高位开始删比后面的数大的。

代码

```
t = int(input())
for i in range(t):
    n,k = input().split()
    n = list(n)
    n = [int(x) for x in n]
    k = int(k)
    stack = []
    for i in n:
        while k>0 and stack and i<stack[-1]:
            k-=1
            stack.pop()
        stack.append(i)
    if k>0:
        stack=stack[:-k]
    stack = [str(x) for x in stack]
    out = ''.join(stack)
    print(out)
```

代码运行截图 (至少包含有"Accepted")

**#50801489提交状态**

---

状态: Accepted

*image-6.png*

用时: 18min左右

## 2. 学习总结和收获

借着月考题重温了一下之前的二分查找、dp、stack等知识点。最近终于过完期中周了，打算开始增加练习。