

Assignment #3: March月考

Updated 1537 GMT+8 March 6, 2024

2024 spring, Compiled by Xinjie Song, Phy

说明:

- 1) The complete process to learn DSA from scratch can be broken into 4 parts:
 - Learn about Time and Space complexities
 - Learn the basics of individual Data Structures
 - Learn the basics of Algorithms
 - Practice Problems on DSA
- 2) 请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora <https://typoraio.cn>，或者用 word）。AC 或者没有AC，都请标上每个题目大致花费时间。
- 3) 提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、“作业评论”区有上传的md或者doc附件。
- 4) 如果不能在截止前提交作业，请写明原因。

编程环境

操作系统: Windows 11 22H2

Python编程环境: PyCharm 2023.2 (Community Edition)

C/C++编程环境: g++ (x86_64-win32-seh-rev0, Built by MinGW-W64 project) 8.1.0

1. 题目

02945: 拦截导弹

<http://cs101.openjudge.cn/practice/02945/>

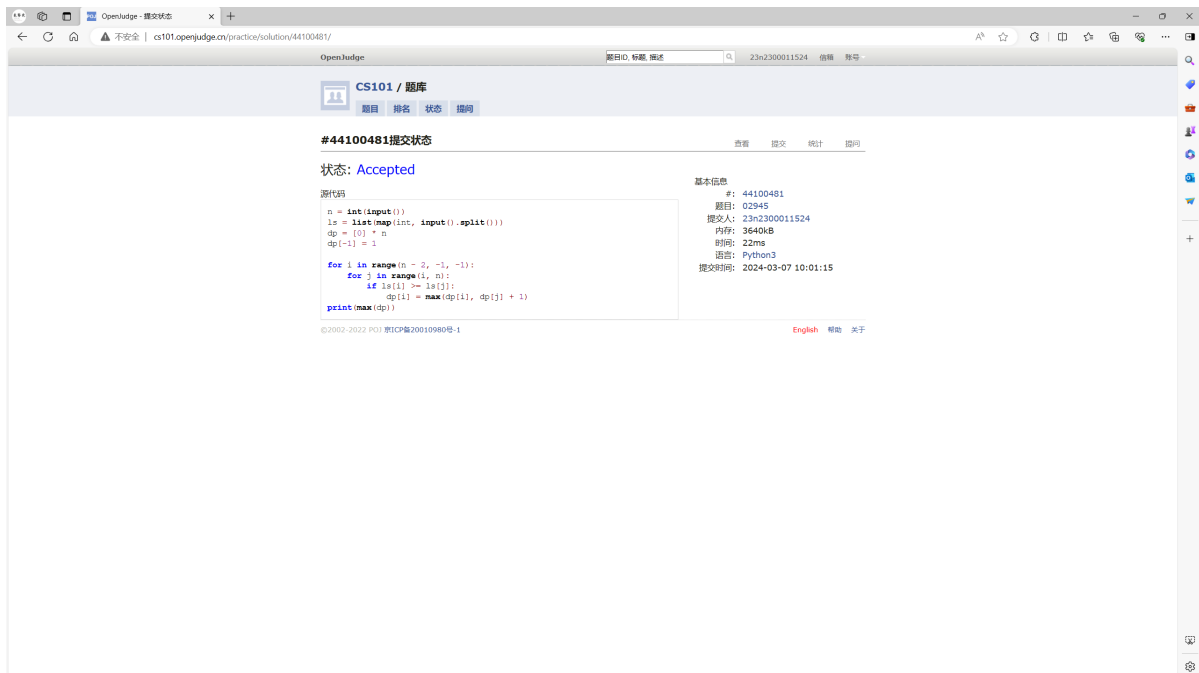
思路: 动态规划

代码

```
n = int(input())
ls = list(map(int, input().split()))
dp = [0] * n
dp[-1] = 1

for i in range(n - 2, -1, -1):
    for j in range(i, n):
        if ls[i] >= ls[j]:
            dp[i] = max(dp[i], dp[j] + 1)
print(max(dp))
```

代码运行截图



04147:汉诺塔问题(Tower of Hanoi)

<http://cs101.openjudge.cn/practice/04147>

思路：递归

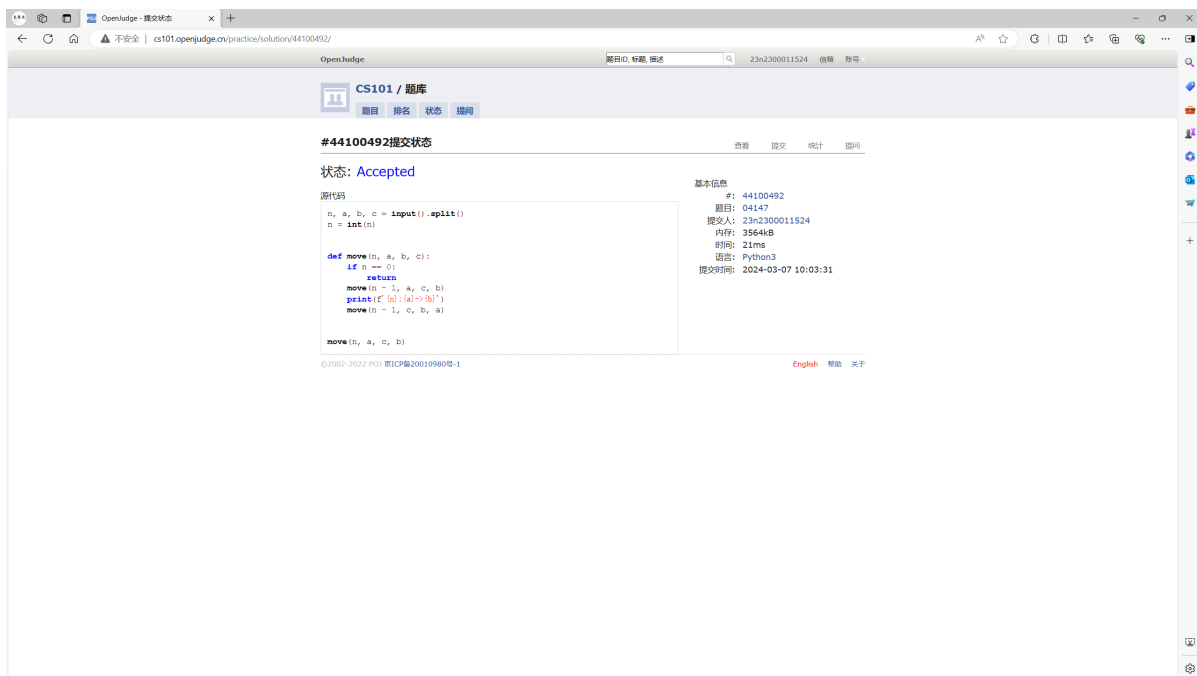
代码

```
n, a, b, c = input().split()
n = int(n)

def move(n, a, b, c):
    if n == 0:
        return
    move(n - 1, a, c, b)
    print(f'{n}:{a}->{b}')
    move(n - 1, c, b, a)

move(n, a, c, b)
```

代码运行截图



03253: 约瑟夫问题No.2

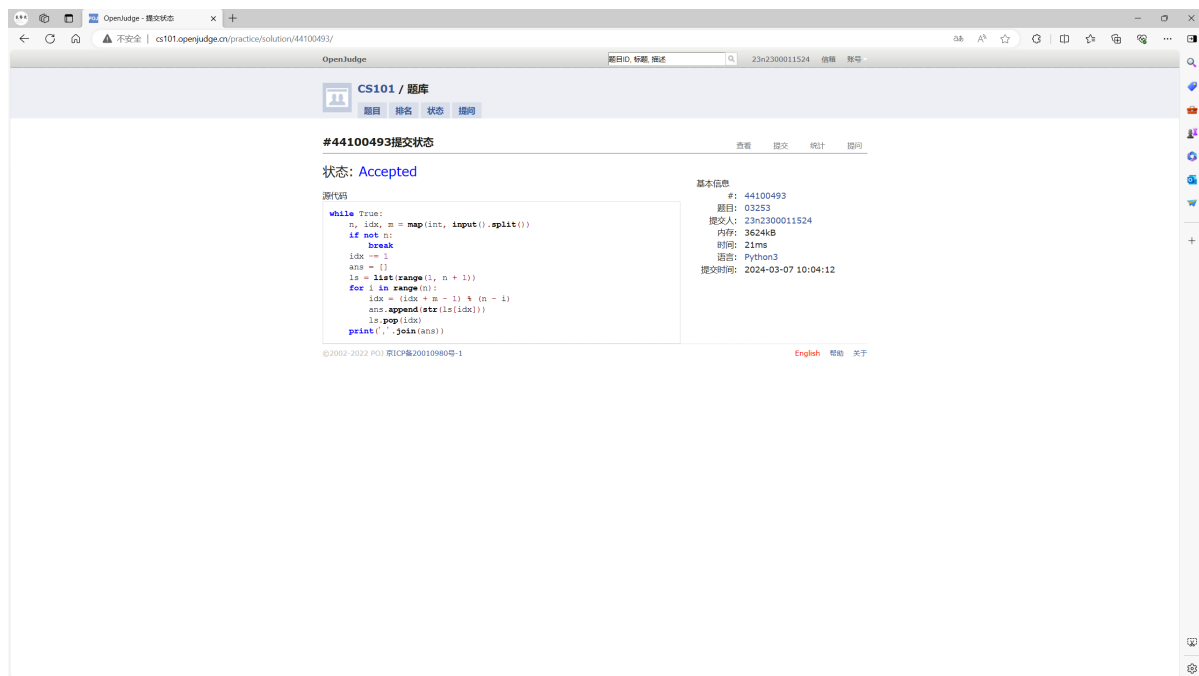
<http://cs101.openjudge.cn/practice/03253>

思路：模拟法

代码

```
while True:
    n, idx, m = map(int, input().split())
    if not n:
        break
    idx -= 1
    ans = []
    ls = list(range(1, n + 1))
    for i in range(n):
        idx = (idx + m - 1) % (n - i)
        ans.append(str(ls[idx]))
        ls.pop(idx)
    print(' '.join(ans))
```

代码运行截图



21554:排队做实验 (greedy)v0.2

<http://cs101.openjudge.cn/practice/21554>

思路：按所需时间排序

代码

```
n = int(input())
ls = list(map(int, input().split()))
ls = sorted([(ls[i], i + 1) for i in range(n)])
time = 0
total_time = 0
for i in range(n):
    total_time += time
    time += ls[i][0]
print(' '.join([str(s[1]) for s in ls]))
print('%.2f' % (total_time / n))
```

代码运行截图



19963:买学区房

<http://cs101.openjudge.cn/practice/19963>

思路：常规思路

代码

```
def mid_num(ls):
    n = len(ls)
    if n % 2 == 0:
        return (ls[n // 2 - 1] + ls[n // 2]) / 2
    return ls[n // 2]

dist_ls = []
```

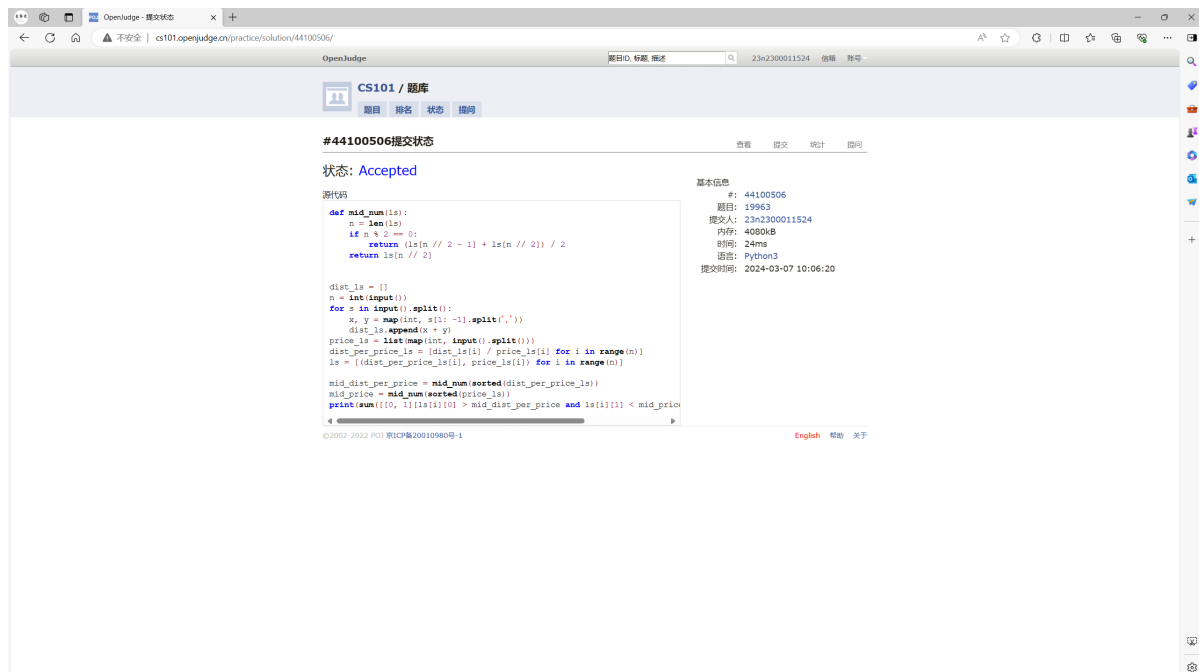
```

n = int(input())
for s in input().split():
    x, y = map(int, s[1:-1].split(','))
    dist_ls.append(x + y)
price_ls = list(map(int, input().split()))
dist_per_price_ls = [dist_ls[i] / price_ls[i] for i in range(n)]
ls = [(dist_per_price_ls[i], price_ls[i]) for i in range(n)]

mid_dist_per_price = mid_num(sorted(dist_per_price_ls))
mid_price = mid_num(sorted(price_ls))
print(sum([[0, 1][ls[i][0] > mid_dist_per_price and ls[i][1] < mid_price] for i
in range(n)]))

```

代码运行截图



27300: 模型整理

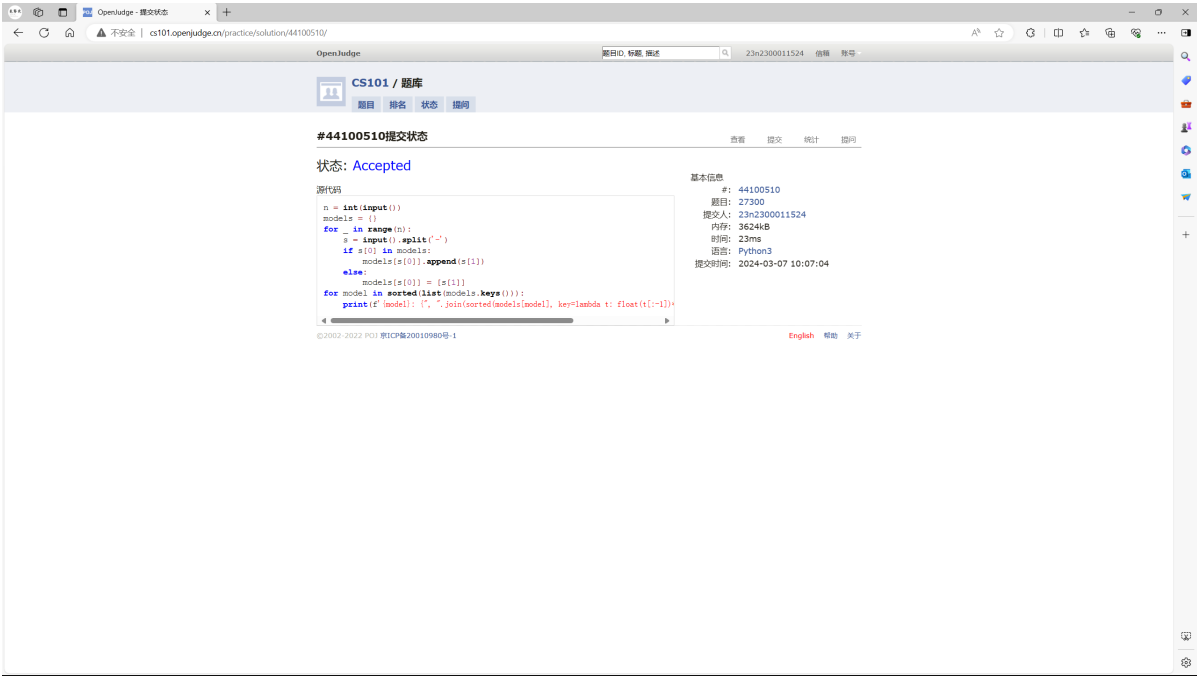
<http://cs101.openjudge.cn/practice/27300>

思路：自定义排序

代码

```
n = int(input())
models = {}
for _ in range(n):
    s = input().split('-')
    if s[0] in models:
        models[s[0]].append(s[1])
    else:
        models[s[0]] = [s[1]]
for model in sorted(list(models.keys())):
    print(f'{model}: {"", ".join(sorted(models[model], key=lambda t: float(t[:-1])*[1, 1000][t[-1] == "B"])))}')#
```

代码运行截图



2. 学习总结和收获

本着复习的原则，考试时重新敲了一遍，用时约40分钟，并没有像某位佬一样按照“数算的精华在于复用”这一原则10分钟速通月考。