# Assignment #D: 十全十美

Updated 1254 GMT+8 Dec 17, 2024

2024 fall, Complied by <mark>同学的姓名、院系</mark>

说明：

1）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。

2）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。

3）如果不能在截止前提交作业，请写明原因。

## 1. 题目

### 02692: 假币问题

brute force, http://cs101.openjudge.cn/practice/02692

思路：

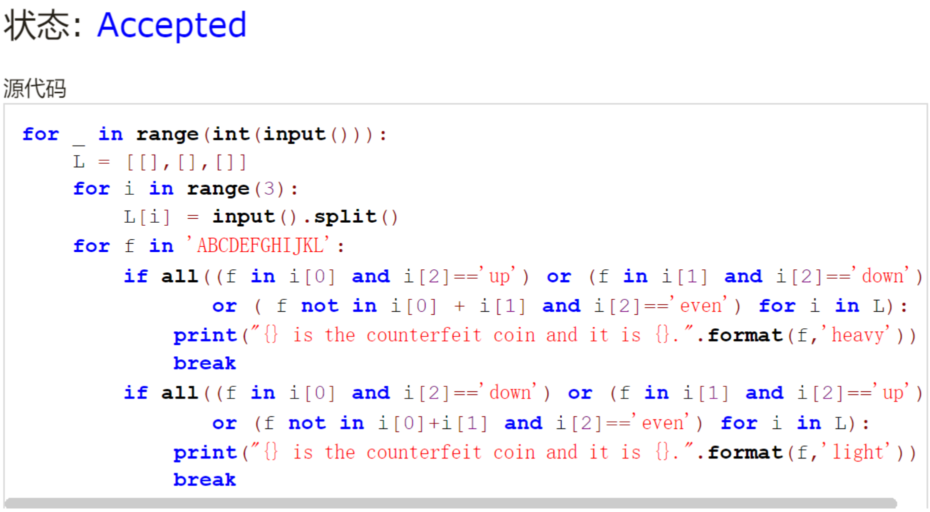
代码：

```python

```

````
for _ in range(int(input())):
    L = [[],[],[]]
    for i in range(3):
        L[i] = input().split()
    for f in 'ABCDEFGHIJKL':
        if all((f in i[0] and i[2]=='up') or (f in i[1] and i[2]=='down')
````

```
                or ( f not in i[0] + i[1] and i[2]==’even’) for i in L):
            print(”{} is the counterfeit coin and it is {}.”.format(f,’heavy’))
            break
        if all((f in i[0] and i[2]==’down’) or (f in i[1] and i[2]==’up’)
                or (f not in i[0]+i[1] and i[2]==’even’) for i in L):
            print(”{} is the counterfeit coin and it is {}.”.format(f,’light’))
            break
```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



状态: Accepted

源代码

```
for _ in range(int(input())):
    L = [[],[],[]]
    for i in range(3):
        L[i] = input().split()
    for f in 'ABCDEFGHIJKL':
        if all((f in i[0] and i[2]=='up') or (f in i[1] and i[2]=='down')
               or ( f not in i[0] + i[1] and i[2]=='even') for i in L):
            print("{} is the counterfeit coin and it is {}.".format(f,'heavy'))
            break
        if all((f in i[0] and i[2]=='down') or (f in i[1] and i[2]=='up')
               or (f not in i[0]+i[1] and i[2]=='even') for i in L):
            print("{} is the counterfeit coin and it is {}.".format(f,'light'))
            break
```

### 01088: 滑雪

dp, dfs similar, http://cs101.openjudge.cn/practice/01088

思路：

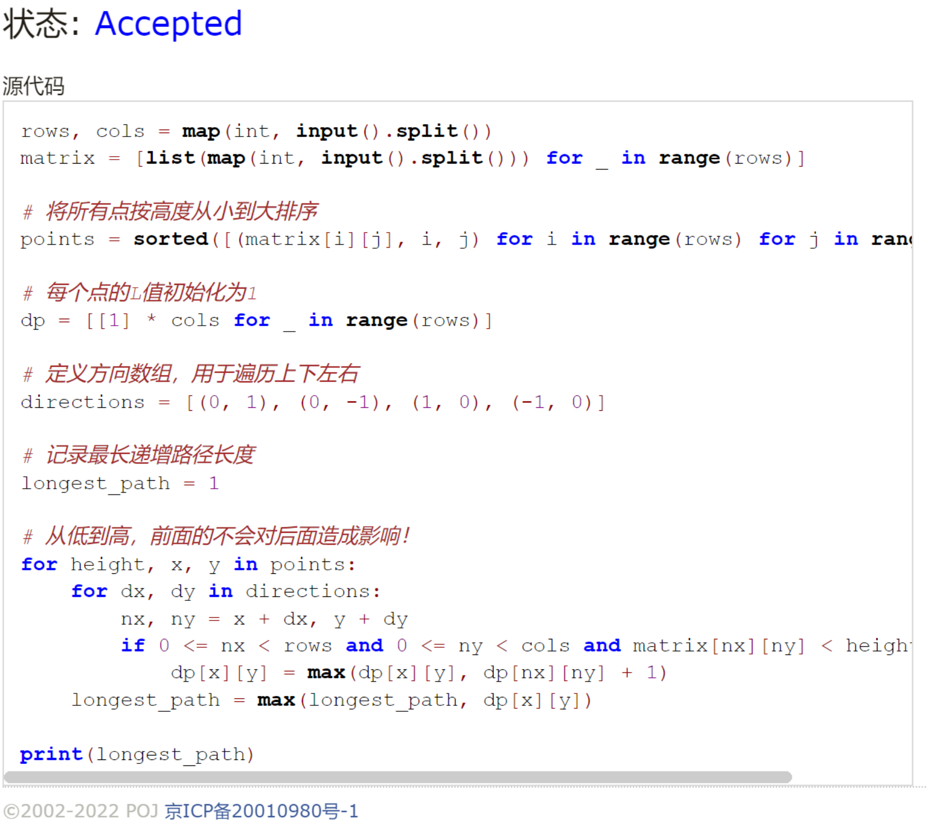
代码：

```python

```

rows, cols = map(int, input().split())

```
matrix = [list(map(int, input().split())) for _ in range(rows)]

# 将所有点按高度从小到大排序
points = sorted([(matrix[i][j], i, j) for i in range(rows) for j in
range(cols)])

# 每个点的 L 值初始化为 1
dp = [[1] * cols for _ in range(rows)]

# 定义方向数组，用于遍历上下左右
directions = [(0, 1), (0, -1), (1, 0), (-1, 0)]

# 记录最长递增路径长度
longest_path = 1

# 从低到高，前面的不会对后面造成影响!
for height, x, y in points:
    for dx, dy in directions:
        nx, ny = x + dx, y + dy
        if 0 <= nx < rows and 0 <= ny < cols and matrix[nx][ny] < height:
            dp[x][y] = max(dp[x][y], dp[nx][ny] + 1)
    longest_path = max(longest_path, dp[x][y])

print(longest_path)
```

代码运行截图 ==（至少包含有"Accepted"）==



状态: Accepted

源代码

```python
rows, cols = map(int, input().split())
matrix = [list(map(int, input().split())) for _ in range(rows)]

# 将所有点按高度从小到大排序
points = sorted([(matrix[i][j], i, j) for i in range(rows) for j in ran

# 每个点的L值初始化为1
dp = [[1] * cols for _ in range(rows)]

# 定义方向数组，用于遍历上下左右
directions = [(0, 1), (0, -1), (1, 0), (-1, 0)]

# 记录最长递增路径长度
longest_path = 1

# 从低到高，前面的不会对后面造成影响！
for height, x, y in points:
    for dx, dy in directions:
        nx, ny = x + dx, y + dy
        if 0 <= nx < rows and 0 <= ny < cols and matrix[nx][ny] < heigh
            dp[x][y] = max(dp[x][y], dp[nx][ny] + 1)
    longest_path = max(longest_path, dp[x][y])

print(longest_path)
```

©2002-2022 POJ 京ICP备20010980号-1

### 25572: 螃蟹采蘑菇

bfs, dfs, http://cs101.openjudge.cn/practice/25572/

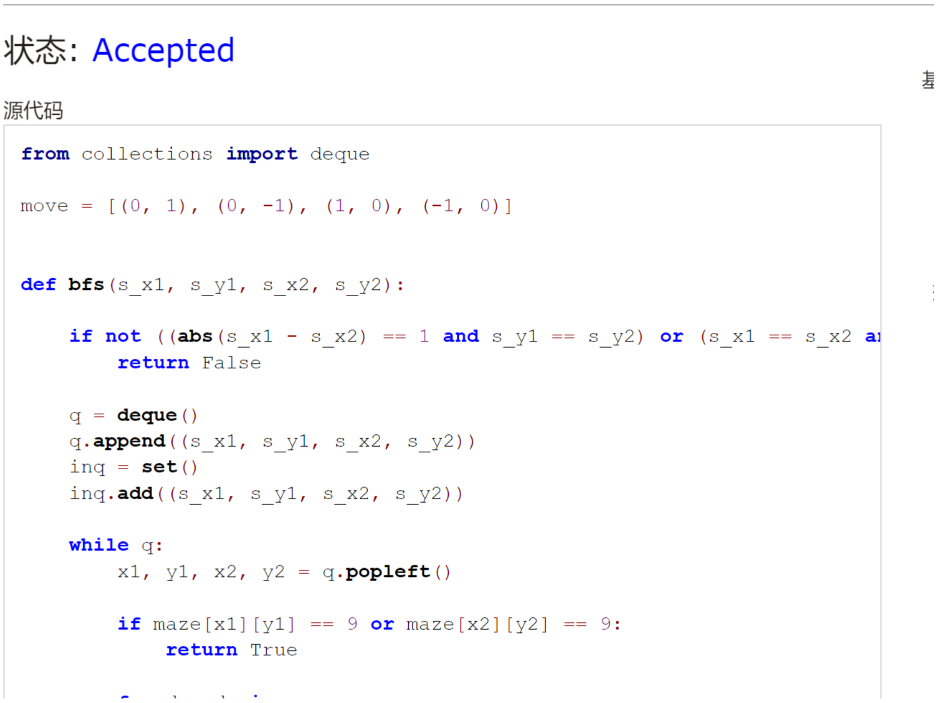
思路:

代码:

```python

```

```
from collections import deque

move = [(0, 1), (0, -1), (1, 0), (-1, 0)]


def bfs(s_x1, s_y1, s_x2, s_y2):

    if not ((abs(s_x1 - s_x2) == 1 and s_y1 == s_y2) or (s_x1 == s_x2
and abs(s_y1 - s_y2) == 1)):
        return False

    q = deque()
    q.append((s_x1, s_y1, s_x2, s_y2))
    inq = set()
    inq.add((s_x1, s_y1, s_x2, s_y2))

    while q:
        x1, y1, x2, y2 = q.popleft()

        if maze[x1][y1] == 9 or maze[x2][y2] == 9:
            return True

        for dx, dy in move:
            nx1, ny1 = x1 + dx, y1 + dy
            nx2, ny2 = x2 + dx, y2 + dy

            if 0 <= nx1 < n and 0 <= ny1 < n and 0 <= nx2 < n and 0 <=
ny2 < n:
                if maze[nx1][ny1] != 1 and maze[nx2][ny2] != 1:
                    if (nx1, ny1, nx2, ny2) not in inq:
                        inq.add((nx1, ny1, nx2, ny2))
                        q.append((nx1, ny1, nx2, ny2))

    return False



n = int(input())
maze = [list(map(int, input().split())) for _ in range(n)]
a = []
for i in range(n):
    for j in range(n):
        if maze[i][j] == 5:
            a.append([i, j])


if len(a) == 2:
    result = bfs(a[0][0], a[0][1], a[1][0], a[1][1])
    print(’yes’ if result else ’no’)
else:
    print(’no’)
```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



状态: Accepted

源代码

```
from collections import deque

move = [(0, 1), (0, -1), (1, 0), (-1, 0)]


def bfs(s_x1, s_y1, s_x2, s_y2):

    if not ((abs(s_x1 - s_x2) == 1 and s_y1 == s_y2) or (s_x1 == s_x2 ar
        return False

    q = deque()
    q.append((s_x1, s_y1, s_x2, s_y2))
    inq = set()
    inq.add((s_x1, s_y1, s_x2, s_y2))

    while q:
        x1, y1, x2, y2 = q.popleft()

        if maze[x1][y1] == 9 or maze[x2][y2] == 9:
            return True
```

### 27373：最大整数

dp，http://cs101.openjudge.cn/practice/27373/

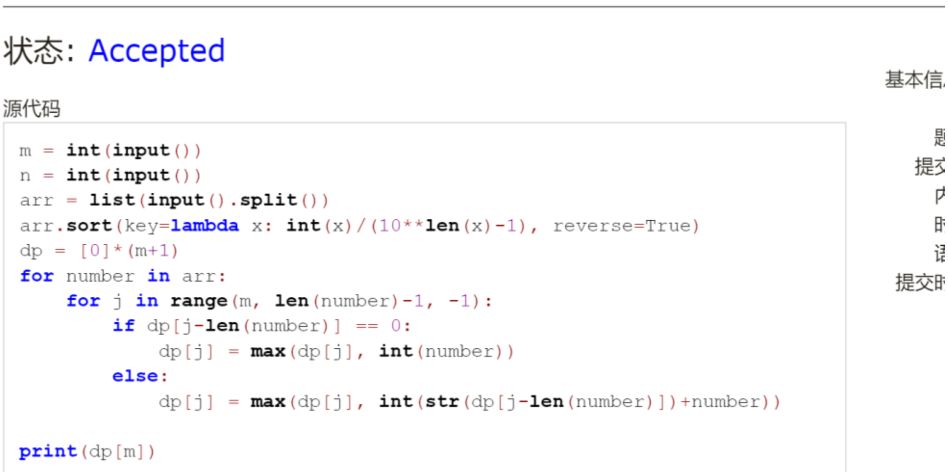
思路：

代码：

```python
```

```
m = int(input())
n = int(input())
arr = list(input().split())
arr.sort(key=lambda x: int(x)/(10**len(x)-1), reverse=True)
dp = [0]*(m+1)
for number in arr:
    for j in range(m, len(number)-1, -1):
        if dp[j-len(number)] == 0:
            dp[j] = max(dp[j], int(number))
        else:
            dp[j] = max(dp[j], int(str(dp[j-len(number)])+number))

print(dp[m])
```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



```
m = int(input())
n = int(input())
arr = list(input().split())
arr.sort(key=lambda x: int(x)/(10**len(x)-1), reverse=True)
dp = [0]*(m+1)
for number in arr:
    for j in range(m, len(number)-1, -1):
        if dp[j-len(number)] == 0:
            dp[j] = max(dp[j], int(number))
        else:
            dp[j] = max(dp[j], int(str(dp[j-len(number)])+number))

print(dp[m])
```

### 02811: 熄灯问题

brute force, http://cs101.openjudge.cn/practice/02811

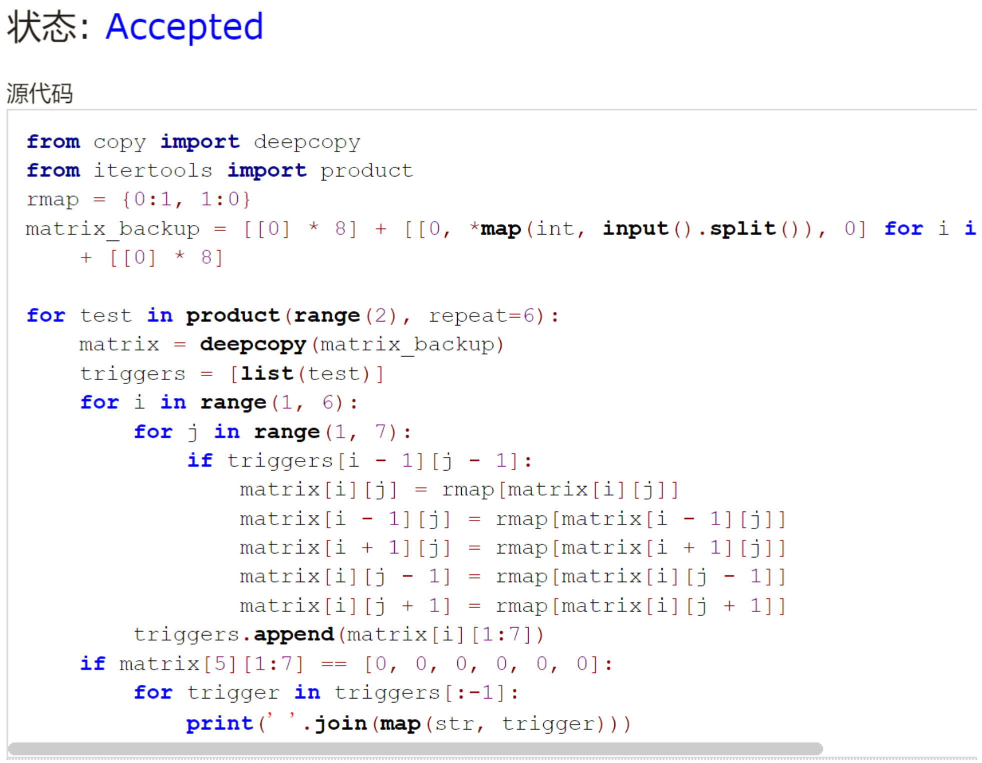
思路:

代码:

```python

```

```
from copy import deepcopy
from itertools import product
rmap = {0:1, 1:0}
matrix_backup = [[0] * 8] + [[0, *map(int, input().split()), 0] for i in range(5)] \
    + [[0] * 8]

for test in product(range(2), repeat=6):
    matrix = deepcopy(matrix_backup)
    triggers = [list(test)]
    for i in range(1, 6):
        for j in range(1, 7):
            if triggers[i - 1][j - 1]:
                matrix[i][j] = rmap[matrix[i][j]]
                matrix[i - 1][j] = rmap[matrix[i - 1][j]]
                matrix[i + 1][j] = rmap[matrix[i + 1][j]]
                matrix[i][j - 1] = rmap[matrix[i][j - 1]]
                matrix[i][j + 1] = rmap[matrix[i][j + 1]]
        triggers.append(matrix[i][1:7])
    if matrix[5][1:7] == [0, 0, 0, 0, 0, 0]:
        for trigger in triggers[:-1]:
            print(' '.join(map(str, trigger)))
```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>



状态: Accepted

源代码

```
from copy import deepcopy
from itertools import product
rmap = {0:1, 1:0}
matrix_backup = [[0] * 8] + [[0, *map(int, input().split()), 0] for i i
    + [[0] * 8]

for test in product(range(2), repeat=6):
    matrix = deepcopy(matrix_backup)
    triggers = [list(test)]
    for i in range(1, 6):
        for j in range(1, 7):
            if triggers[i - 1][j - 1]:
                matrix[i][j] = rmap[matrix[i][j]]
                matrix[i - 1][j] = rmap[matrix[i - 1][j]]
                matrix[i + 1][j] = rmap[matrix[i + 1][j]]
                matrix[i][j - 1] = rmap[matrix[i][j - 1]]
                matrix[i][j + 1] = rmap[matrix[i][j + 1]]
        triggers.append(matrix[i][1:7])
    if matrix[5][1:7] == [0, 0, 0, 0, 0, 0]:
        for trigger in triggers[:-1]:
            print(' '.join(map(str, trigger)))
```

### 08210：河中跳房子

binary search, greedy, http://cs101.openjudge.cn/practice/08210/

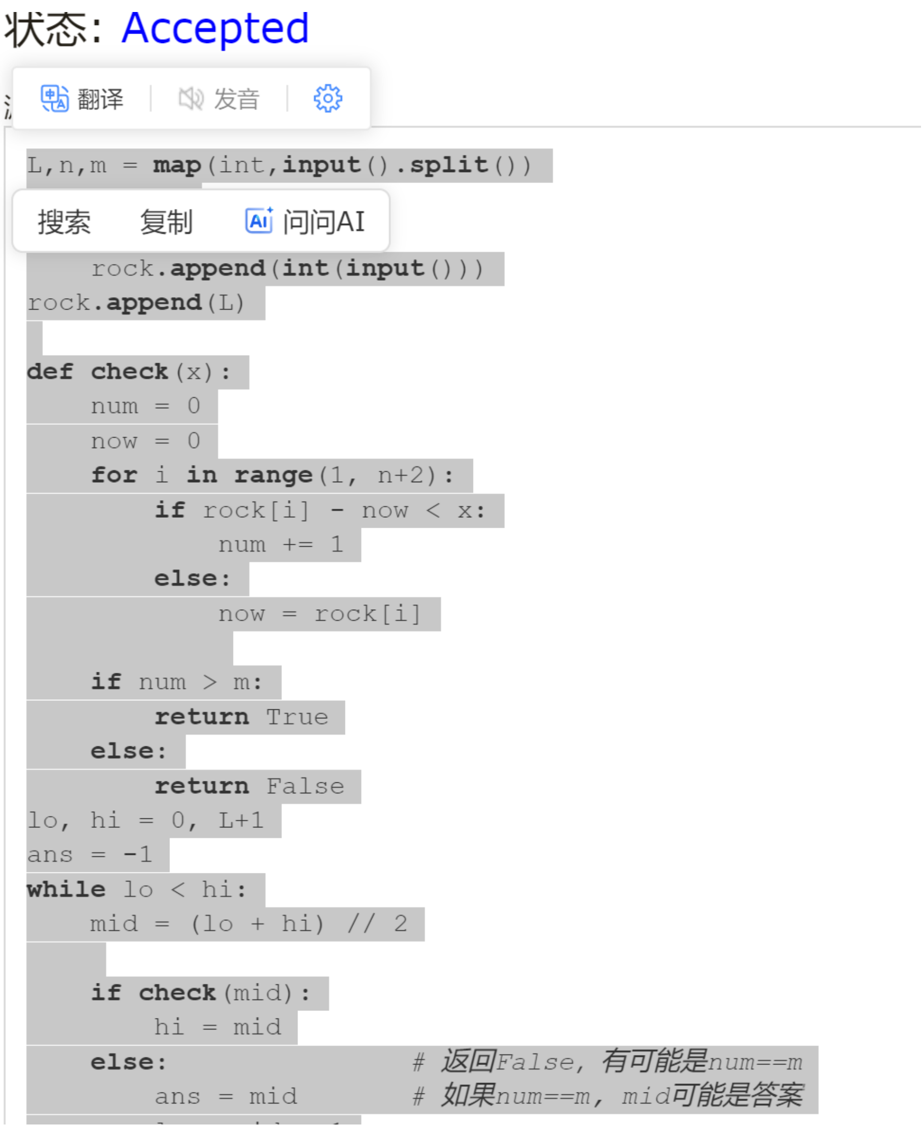
思路：

代码：

```python

```

代码运行截图 <mark>（至少包含有"Accepted"）</mark>

```
L,n,m = map(int,input().split())
rock = [0]
for i in range(n):
    rock.append(int(input()))
rock.append(L)

def check(x):
    num = 0
    now = 0
    for i in range(1, n+2):
        if rock[i] - now < x:
            num += 1
        else:
            now = rock[i]

    if num > m:
        return True
    else:
        return False
lo, hi = 0, L+1
ans = -1
while lo < hi:
    mid = (lo + hi) // 2

    if check(mid):
        hi = mid
    else:               # 返回False，有可能是num==m
        ans = mid       # 如果num==m, mid可能是答案
        lo = mid + 1

#print(lo-1)
print(ans)
```



状态: Accepted

```
L,n,m = map(int,input().split())

    rock.append(int(input()))
rock.append(L)

def check(x):
    num = 0
    now = 0
    for i in range(1, n+2):
        if rock[i] - now < x:
            num += 1
        else:
            now = rock[i]

    if num > m:
        return True
    else:
        return False
lo, hi = 0, L+1
ans = -1
while lo < hi:
    mid = (lo + hi) // 2

    if check(mid):
        hi = mid
    else:               # 返回False，有可能是num==m
        ans = mid       # 如果num==m, mid可能是答案
```

## 2. 学习总结和收获

<mark>如果作业题目简单，有否额外练习题目，比如：OJ“计概 2024fall 每日选做”、CF、LeetCode、洛谷等网站题目。</mark>

机考加油，希望题目别让我对这门课彻底失望