

goorm

코딩 테스트 해설 강의

강의자 : 권우영(ilium)



1번

해결 방법 찾기



1번

코드 작성하기

```
import sys
input = sys.stdin.readline

N, M = map(int, input().split())
# N의 최대가 100이기 때문에 일반적인 배열로 선언
box = [0 for _ in range(N+1)]

for _ in range(M):
    boxNumber, command, ballCnt = map(int, input().split())

    if command == 1:
        box[boxNumber] += ballCnt
    else:
        if box[boxNumber] > ballCnt:
            box[boxNumber] -= ballCnt

for i in range(1, N):
    print(box[i])
```



2번

해결 방법 찾기



2번

정해 코드 작성

```
import sys
input = sys.stdin.readline

N, K = map(int, input().split())
inven = list()
for _ in range(N):
    value, count = map(int, input().split())
    inven.append([value, count])

# 그리디한 선택을 위해서 정렬함.
inven.sort(key=lambda x : x[0], reverse=True)
answer = 0
```



2번

정해 코드 작성

```
for i in range(N):  
    if K > 0: # 더블업 할 수 있는 수량이 남아 있는 경우  
        if K >= inven[i][1]:  
            # 일부 예외 처리  
            answer += inven[i][0] * 2 * inven[i][1]  
            K -= inven[i][1]  
        else:  
            answer += inven[i][0] * 2 * (K)  
            answer += inven[i][0] * (inven[i][1]-K)  
            K = 0  
    else: # 수량이 남아 있지 않는 경우  
        answer += inven[i][0] * inven[i][1]  
  
print(answer)
```



3번

해결 방법 찾기



3번

정해 코드 작성

```
N, M = map(int, input().split())  
A = set(map(str, input().split())) # 탐색 속도를 O(1) 로 줄이는 전략  
B = set(map(str, input().split()))
```

```
for _ in range(M):  
    chA, chB = map(str, input().split())  
    if chA in A and chB in B:  
        A.remove(chA)  
        B.remove(chB)  
        A.add(chB)  
        B.add(chA)
```

```
A = list(A)  
A.sort()  
print(*A)
```



4번

해결 방법 찾기

4번

정해코드 작성

```
import sys
from collections import deque
input = sys.stdin.readline

N = int(input())
matrix = list()
for _ in range(N):
    matrix.append(list(map(int, input().split())))

# 방문 기록을 저장할 변수
visited = [[False for _ in range(N)] for _ in range(N)]
# 최대 도형 수를 저장할 변수
result = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

dx = [0, 0, 1, -1]
dy = [1, -1, 0, 0]
```



4번

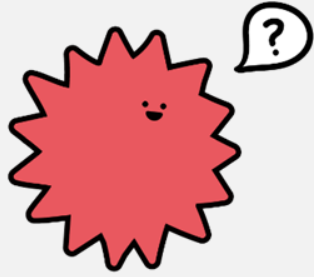
정해코드 작성

```
# 탐색 시작 위치 찾기
for i in range(N):
    for j in range(N):
        if not visited[i][j]:
            target = matrix[i][j]
            q = deque()
            q.append([i, j])
            visited[i][j] = True
            cnt = 1 # 매 도형 수의 크기를 측정
```

```
# 탐색 시작
while q:
    cx, cy = q.popleft()
    for k in range(4):
        nx = cx + dx[k]
        ny = cy + dy[k]
        if 0 <= nx < N and 0 <= ny < N:
            if not visited[nx][ny] and matrix[nx][ny] == target:
                q.append([nx, ny])
                visited[nx][ny] = True
                cnt += 1

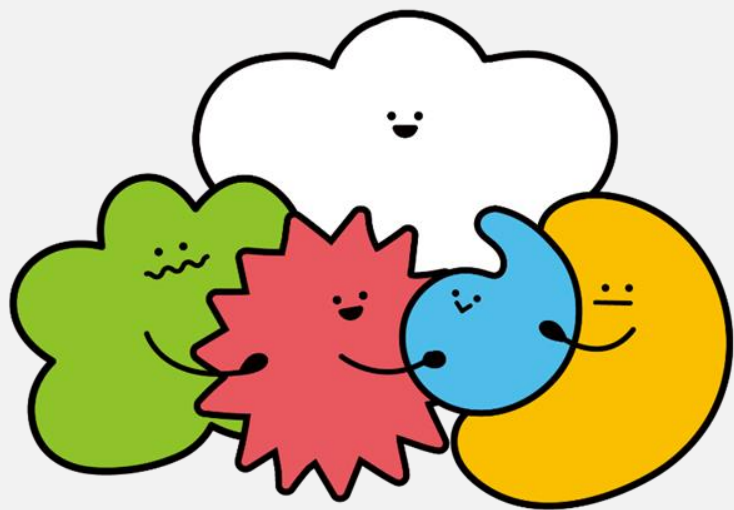
# 최대 도형 수 갱신
if result[target] < cnt:
    result[target] = cnt
answer = ''.join(map(str, result[1:]))
print(answer)
```





Q&A





goorm