

동적테이블 활용

광석 수집

출발>도착 지점

오른쪽, 아래로만 움직이면서 광석이 있는 칸을 최대한 많이 거쳐 지나갈 때, 광석의 개수를 출력

위, 왼쪽에서 진입하는 경우만 존재하기 때문에 dp테이블의 x-1, y-1의 칸의 경우의수를 고려하면 된다.

재귀함수 구현 시 주의점

1. 종료조건 : 도착지점에 도달했을 때
2. 경계조건 : 맵의 크기를 벗어날 때

$DT[1][1] = A[1][1];$

$DT[a][b] = \text{Max}(DT[a-1][b], DT[a][b-1]) + A[a][b];$

```
for(int i=1;i<=n;i++){
    for(int j=1;j<=m;j++){
        if(i==1&&j==1) DT[i][j] = A[i][j];
        else{
            DT[i][j] = Max(DT[i-1][j], DT[i][j-1]) + A[i][j];
        }
    }
}
```

색상환문제

7

선물

모든 경우를 검사해야하는 전체탐색부분의 연산량을 줄이는데 이용할 수 있다

선물을 3명에게 분배하는 문제

조건 1 : 각자의 부피는 $길동 \geq 길순 \geq 길삼$

조건 2 : d가 최소가 되게 한다. $d = \text{길동선물의 부피합} - \text{길삼선물의 부피합}$

조건 3 : d가 배분되는 방법이 여럿일 경우에는 길동의 선물의 부피합이 적은 방법을 선택한다.

예) 6 4 4 4 6 9

길동 : 12 길순 : 12 길삼 : 9(답)

길동 : 13 길순 : 10 길삼 : 10 - 조건 3 위배

전체탐색 : 선물 하나당 길동, 길삼, 길순의 경우를 다 검사함

동적테이블 : $W = a+b+c$

1-k개의 선물을 배분을 했을 때, 선물 부피의 총합

= 결정문제 k개의 선물을 배분했을 때, 길순이가 100 길삼이가 76인 상태가 존재할 수 있는가? > yes

or no

예) 6 4 4 4 6 9

1번째 : $W=6, b=6$ or $c=6$ or $a = 6-(b+c)$

동적 테이블

열 : 길삼이가 받은 선물의 부피합

행 : 길순이가 받은 선물의 부피합

$DT[a][b]=1$

첫번째 선물

길동 : $DT[0][0] = \text{true};$

길순 : $DT[6][0] = \text{true};$

길삼 : $DT[0][6] = \text{true};$

두번째 선물 : 4

이전 테이블에 true인 값에 행+4, 열+4 부분을 true로 바꿈

답 : 마지막 테이블에 저장된 true라면 조건 1, 2, 3을 검사하면서 answer값을 갱신해감

두부모판 자르기

$N \times n$ 두부 모판을

1×2 또는

2×1 크기로 잘라서 판매한다

잘라진 모판은 등급표를 이용해 이익을 계산할 수 있다. 주어진 모판에서 얼마나 최대의 이익을 얻을 수 있는지 구하는 문제

선택지

현재 a행, b열 에서

1. $(a, b) (a+1, b)$ 로 판매 - 세로
2. $(a, b) (a, b+1)$ 로 판매 - 가로
3. 판매하지 않음