

6. 정보기술관의 이상한 엘리베이터

시간 제한 : 2초 | 메모리 제한 : 256MB

해설

수열의 길이 N은 최대 500, 건물의 층 수 H는 최대 1000이다.

주어진 수열의 각 자리에서 아래로 가는 방법과 위로 가는 방법 2가지 모두 가능하다고 가정하면, 일반적인 재귀함수로 작성할 경우 수열의 길이가 최대 500 이므로

$2 * 2 * 2 * \dots$ = 최대 2500번의 계산이 필요 (Brute-Force - Time-limit Exceeded)

동적계획법(Dynamic Programming)으로 문제 해결해야 한다.

$dp[idx][floor]$: idx 번째 수열일 때, floor층을 갈 수 있는지 여부를 확인하는 dp table 생성.

ex)

만약 수열이 10 5 7이라면

$dp[0][10] = \text{true} \rightarrow dp[1][15] = \text{true} \rightarrow dp[2][22] = \text{true}$

↘ $dp[2][8] = \text{true}$

↘ $dp[1][5] = \text{true} \rightarrow dp[2][12] = \text{true}$

↘ $dp[2][-2]$ (불가능)

주의할 점

현재 층 + 수열의 값 $\leq H$, 현재 층 - 수열의 값 ≥ 0 인지 확인해야함.

시간 복잡도는 $dp[500][1000]$ 을 채우는 것과 같으므로 $O(NM)$ 이다.

N 최대 500, H 최대 1000 이므로 $500 \times 1000 = 500,000$ 이므로 제한 시간안에 가능!

dp를 채우고 마지막 수열에서 갈수 있는 최대 층 수를 찾아 정답 조건에 맞게 출력.

** Bottom-Up 방식과 Top-Down 방식 둘다 가능

동일한 문제이므로 블로그 소스 코드 참고하시면 됩니다.

Top-Down 방식 : <http://vjerksen.tistory.com/41> [vjerksen](#) 님 블로그

Bottom-Up 방식 : <http://codersbrunch.blogspot.kr/2016/12/1495.html> [choyi0521](#) 님 블로그