

알고리즘 초보자들을 위한 기초자료 - 4

SimJoon_TTMD

2670

- 누적곱으로 풀다가 틀렸다.
- $Dp[i]$ = i 번째 까지의 값을 사용했을 때 나오는 최대로 고쳐서 풀면 빨리 풀린다

SimJoon_TTMD

2533

- $Dp[i][0,1]$ 을 i 를 루트 노드로하는 서브트리로 만들었을 때 필요한 얼리어답터의 값이라고 정의하자.
- $Dp[i][0]$ 은 i 번째 노드를 얼리어답터가 아니라고 했을 때고,
- $Dp[i][1]$ 은 i 번째 노드를 얼리어답터라고 했을 때다.
- 모든 노드는 루트노드가 가능하기에 편의상 1을 루트노드로하고 풀었다

SimJoon_TTMD

1932

- $Dp[i][j]$ = 위에서 i 번째, 왼쪽에서 j 번째 수에서의 최대값
- $Dp[i][j] = (\max(dp[i-1][j], dp[i-1][j-1]) + arr[i][j])$ 이다
- 자기 왼쪽상단 혹은 오른쪽 상단만 선택 가능하기 때문이다

SimJoon_TTMD

1890

- 아래 혹은 오른쪽으로만 이동이 가능하기에
- $Arr[i][j]$ 에서는 $Arr[i+arr[i][j]][j]$ 혹은 $arr[i][j+arr[i][j]]$ 로 이동 가능하다

SimJoon_TTMD

1309

- $Dp[i] = dp[i-1]*2 + dp[i-2]$
- 2*1 타일문제로 바뀌어서 생각해도 된다.
- $Dp[i-1]*2$ 는 위에 한칸 만들고 뒤집는 경우다.