풀이과정

문제 이미지를 보면 삼각형으로 보이지만, 실제로 2차원 배열에 넣어보면 계단 형식으로 값이 들어간다.(1번째는 1개, 2번째는 2개, 3번째는 3개….) 아래층에 있는 수는 현재 층에서 선택된 수의 대각선 왼쪽 또는 대각선 오른쪽에 있는 것 중에서만 선택할 수 있다고 했으므로 (i,j)위치에서의 누적 합의 최대는 (i-1,j-1)까지의 누적합과 (i-1,j+1)까지의 누적합 중 큰 값에 (i,j)를 더한 값이 된다.

1. j==0인 경우 -1 범위는 존재하지 않는다.

Dp[i][0]=dp[i-1][1]+arr[i][0];

1. 0<j<9인 경우 좌우 값을 비교해 최대값을 가져온다.

Dp[i][j]=max(dp[i-1][j-1],dp[i-1][j-1])+arr[i][j];

1. j==9인 경우 10 범위는 존재하지 않는다.

Dp[i][9]=dp[i-1][8]+arr[i][9];

누적합을 구하기 위한 2차원 배열 dp와 삼각형의 각 칸에 들어있는 값을 저장할 arr을 선언하고 두번째 줄부터(첫번째 줄은 dp[0][0]=arr[0][0]밖에 없으므로 초기에 설정해둔다.) 마지막 줄까지 dp를 계산하고 마지막 줄의 dp에서 최대값을 찾아 출력한다.

