##### 멀리 뛰기

Level 3

효진이는 멀리 뛰기를 연습하고 있습니다. 효진이는 한번에 1칸, 또는 2칸을 뛸 수 있습니다. 칸이 총 4개 있을 때, 효진이는  
(1칸, 1칸, 1칸, 1칸)  
(1칸, 2칸, 1칸)  
(1칸, 1칸, 2칸)  
(2칸, 1칸, 1칸)  
(2칸, 2칸)  
의 5가지 방법으로 맨 끝 칸에 도달할 수 있습니다. 멀리뛰기에 사용될 칸의 수 n이 주어질 때, 효진이가 끝에 도달하는 방법이 몇 가지인지 출력하는 jumpCase 함수를 완성하세요. 예를 들어 4가 입력된다면, 5를 반환해 주면 됩니다.

**class JumpCase {**

**public int jumpCase(int num) {**

**int answer = 0;**

**int f1 = 1;**

**int f2 = 2;**

**for(int i = 3 ; i <= num ; i++){**

**answer = f1+f2;**

**f1=f2;**

**f2=answer;**

**}**

**return answer;**

**}**

**public static void main(String[] args) {**

**JumpCase c = new JumpCase();**

**int testCase = 4;**

**//아래는 테스트로 출력해 보기 위한 코드입니다.**

**System.out.println(c.jumpCase(testCase));**

**}**

**}**

**class** **JumpCase** {

**public** **int** jumpCase(**int** num) {

**int** answer = **0**;

**int** dp**[][]** = **new** **int**[**2**][num];

dp[**0**][**0**] = **1**;

dp[**0**][**1**] = **1**;

dp[**1**][**0**] = **0**;

dp[**1**][**1**] = **1**;

**for** (**int** i = **0**; i < **2**; i++) {

**for** (**int** j = **2**; j < num; j++) {

**if** (i == **0**) {

dp[i][j] = dp[i][j - **1**] + dp[i + **1**][j - **1**];

dp[i + **1**][j] = dp[i + **1**][j - **2**] + dp[i][j - **2**];

}

}

}

answer = dp[**1**][num - **1**] + dp[**0**][num - **1**];

**return** answer;

}

**public** **static** **void** main(**String[]** args) {

JumpCase c = **new** JumpCase();

**int** testCase = **4**;

**System**.out.println(c.jumpCase(testCase));

}

}