//문제 1

function solution1() {

let num = gets().trim();

//재귀함수 이용

function fibo(a) {

if (a <= 1) return a;

else return fibo(a - 1) + fibo(a - 2);

}

print(fibo(num));

}

//문제 2

function solution2() {

let a = gets().split(' ');

//유클리드 호제법 이용

function euclid(n, m) {

let r = n % m;

if (r == 0) return m;

else return euclid(m, r);

}

print(euclid(a[0], a[1]));

}

//문제 3

//좋지 않은 방법

//이항 계수의 성질을 이용함

function solution3() {

let a = Number(gets());

let answer = 0;

answer = Math.pow(2, a) - 1;

print(answer);

}

//문제 4

function solution4() {

let a = gets().split(' ');

let answer = 0;

let k = Number(a[1]);

//계단을 오를 수 있는 MAX까지 for문을 돌려, 재귀함수를 실행한다.

function recs(n) {

if (n == 0) {

answer++;

return;

} else if (n < 0) {

return;

} else {

for (let i = 1; i <= k; i++) {

recs(n - i);

}

}

}

recs(Number(a[0]));

print(answer);

}