

Candyland

Problem Statement:

Tom has a sweet tooth, and he frequently visits his local candy shop to buy N pieces of his favourite candy. Tom notices that the scoop used by the shopkeeper is rather small, and he can only take up a maximum of 4 candies per scoop. Tom is very interested in combinatorics; hence, Tom wants to count out the number of distinct ways to scoop up all N candy pieces, assuming that the storekeeper does not make empty scoops. Note that a distinct sequence of scoops must include at least one nonempty scoop. Would you be able to help him?

Input:

There is one line of input, containing a single integer N ($0 \leq N \leq 12$).

Output:

Print the number of distinct ways to scoop up all N candy pieces.

Sample Input 1:

3

Sample Output 1:

4

Sample Input 2:

4

Sample Output 2:

8

Explanation:

When $N = 3$, there are 4 possible distinct ways in which the N candies can be scooped up:

[1,1,1], [1,2], [2,1], [3]

When $N = 4$, there are 8 possible distinct ways in which the N candies can be scooped up:

[1,1,1,1], [1,2,1], [1,1,2], [1,3], [2,1,1], [2,2], [3,1], [4]

Extension:

Can you modify your code such that it will still pass the time limit for large values of N ($N \geq 1000000$)?