B. Buttons

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Manao is trying to open a rather challenging lock. The lock has n buttons on it and to open it, you should press the buttons in a certain order to open the lock. When you push some button, it either stays pressed into the lock (that means that you've guessed correctly and pushed the button that goes next in the sequence), or all pressed buttons return to the initial position. When all buttons are pressed into the lock at once, the lock opens.

Consider an example with three buttons. Let's say that the opening sequence is: {2, 3, 1}. If you first press buttons 1 or 3, the buttons unpress immediately. If you first press button 2, it stays pressed. If you press 1 after 2, all buttons unpress. If you press 3 after 2, buttons 3 and 2 stay pressed. As soon as you've got two pressed buttons, you only need to press button 1 to open the lock.

Manao doesn't know the opening sequence. But he is really smart and he is going to act in the optimal way. Calculate the number of times he's got to push a button in order to open the lock in the worst-case scenario.

Input

A single line contains integer n ($1 \le n \le 2000$) — the number of buttons the lock has.

Output

In a single line print the number of times Manao has to push a button in the worst-case scenario.

Examples

input
2
output
3

input
3
output
7

Note

Consider the first test sample. Manao can fail his first push and push the wrong button. In this case he will already be able to guess the right one with his second push. And his third push will push the second right button. Thus, in the worst-case scenario he will only need 3 pushes.

→ Attention

Package for this problem was not updated by the problem writer or Codeforces administration after we've upgraded the judging servers. To adjust the time limit constraint, solution execution time will be multiplied by 2. For example, if your solution works for 400 ms on judging servers, then value 800 ms will be displayed and used to determine the verdict.

Codeforces Round #164 (Div. 2)

Finished

→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Problem tags

implementation math

No tag edit access

→ Contest materials

- Announcement
- Tutorial