

Problem D

How Many Trees?

Input: standard input

Output: standard output

Memory Limit: 32 MB

A binary search tree is a binary tree with root **k** such that any node **v** in the left subtree of **k** has **label (v) < label (k)** and any node **w** in the right subtree of **k** has **label (w) > label (k)**.

When using binary search trees, one can easily look for a node with a given label **x**: After we compare **x** to the label of the root, either we found the node we seek or we know which subtree it is in. For most binary search trees the average time to find one of its **n** nodes in this way is $O(\log n)$.

Given a number **n**, can you tell how many different binary search trees may be constructed with a **set** of numbers of size **n** such that each element of the set will be associated to the label of exactly one node in a binary search tree?

Input and Output

The input will contain a number $1 \leq i \leq 1000$ per line representing the number of elements of the set. You have to print a line in the output for each entry with the answer to the previous question.

Sample Input

1
2
3

Sample Output

1
2
5

(The Joint Effort Contest, Problem setter: Rodrigo Malta Schmidt)