Cartesian Tree

Input file:	`standard input`
Output file:	`standard output`
Time limit:	0.5 seconds
Memory limit:	256 megabytes

There is a unique way to build a cartesian tree on a set of points (with pairwise distinct x-coordinates and pairwise distinct y-coordinates). A boy throws n points into a 1×1 square randomly and inserts them into the cartesian tree (which is initially empty). We will say that a *rebuilding* took place if the newly-inserted vertex is not a leaf in the new tree. You have to count the expected number of rebuildings.

Since the probability of some two points having same x's or same y's is zero, we can assume that all of them are distinct.

Input

The only line of input contains one integer n ($1 \leq n \leq 10^9$).

Output

Print the answer to the problem with absolute or relative error at most 10^{-9}.

Examples

standard input	standard output
2	0.5000000000
5	2.2388888888

Note

Cartesian tree is a data structure which unites a binary search tree and a binary heap.

Strictly speaking, a cartesian tree is a binary tree such that each of its nodes stores a pair (x, y), with x being the *key*, and y being the *priority*. The cartesian tree must be a binary search tree by x's and a binary heap by y's. If we assume that all x's and all y's are distinct, we conclude that for an element (x_0, y_0) all of its left descendants have $x < x_0$, all of its right descendants have $x > x_0$, and all its descendants must have $y < y_0$. (Definition is translated from ITMO Wikipages).