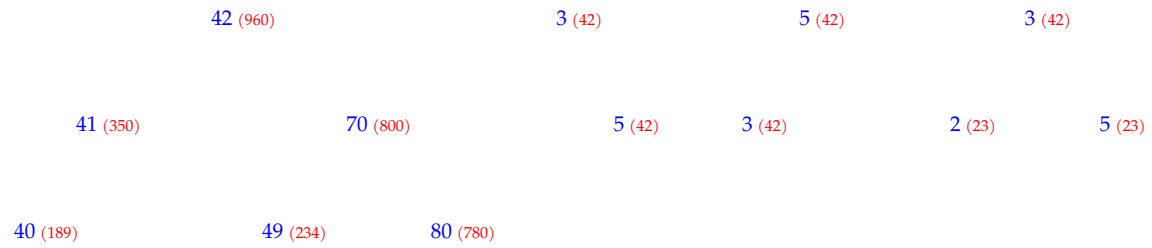
Treap ambiguity — P30477_en

Roughly speaking, a treap is a BST (a binary search tree) and also a heap. Assume that we have n different keys, and a fixed integer m. Every key is independently associated with a priority (a random integer chosen uniformly from the interval $[1, m]$). Let x be the key with the largest priority. Then, the treap for the set of pairs (key, priority) is the BST with x at its root, the treap for the pairs with keys smaller than x as its left subtree, and the treap for the pairs with keys larger than x as its right subtree.

For instance, suppose that the keys are $\{40, 41, 42, 49, 70, 80\}$ and $m = 1000$. Then, the pairs (key, priority) could be (40, 189), (41, 350), (42, 960), (49, 234), (70, 800) and (80, 780). In this case, we would have the treap to the left:



If we only look at the keys (in blue), we have a BST. If we only look at the priorities (in red), we have a sort of a heap (the largest at the top; the same property holds recursively).

This definition is usually good enough in many practical situations. However, there is a caveat: With repeated priorities, we can have more than one possible treap. For example, the second and the third treaps above are possible for the pairs (3, 42) and (5, 42).

Note that we can have a unique treap with repeated priorities. Consider for instance the pairs (2, 23), (3, 42) and (5, 23). The only possible treap is the fourth above.

Given n and m, what is the probability that the treap is unique?

Input

Input consists of several cases, each with n and m. Assume $1 \leq n \leq 50$ and $2 \leq m \leq 5000$.

Output

For every case, print the probability that the treap is unique with four digits after the decimal point. The input cases have no precision issues.

Sample input 1

```
1 100
2 2
3 3
50 5000
```

Sample output 1

```
1.0000
0.5000
0.3333
0.9656
```

Problem information

Author: Salvador Roura

Generation: 2026-01-25T10:09:32.326Z