Stand Up

Problem code: STANDUP

A classroom at the Academy of Bit Wizardry contains **N rows** of chairs. Each row, contains exactly **M chairs**. Each chair is expected to be filled by exactly one student; you may assume no chair is ever empty.

Curiously, **N*M** is guaranteed to be an **even number**. This means there are always an **even number of students** in a class. The students come to the Academy of Bit Wizardry from **(N*M / 2)** coutries. In fact, **exactly 2 students come from each country**.

You enter the classroom and find all the students sitting. This displeases you and you request them to stand up. They tell you that they are going to play a game with you. You will play the game **move by move**. In each move you **first make 1 student stand up and declare his country name**. Then, you make **another student stand up and declare his country name**. If both the students belong to the same country, they both remain **standing**. Otherwise, they will now both sit down. Of course, you may ask the same student to stand up again in a later move.

You grin as you know you have an amazing memory. Although you don't know initially which student belongs to what country, you are sure that **once a student declares it, you will remember it indefinitely**. You will never have to make a student stand up more than twice. Soon, you will have the entire class standing.

There is no way for you to know what country a student belongs to initially. You could choose a fixed strategy that does not optimize itself based on the responses the students are giving and easily have the class standing in exactly **N*M** moves. But you are smarter; hence, you will **dynamically optimize your strategy based on their responses**.

Assuming that the students choose their chairs uniformally randomly over all possible seatings, what is the **expected number of moves** that such a dynamically optimized optimal strategy would take to make every student stand up?

Input

There are at most **100 test cases** in each test file. Each test case contains a single line with two positive integer **N** and **M**. The input is terminated by **EOF**.

Output

For every test case, output a single line with the **expected number of moves** you will take to make every student stand up if you employ an optimal strategy. Print this number **rounded to 4 decimal places**.

Constraints

1 ≤ N ≤ 50
1 ≤ M ≤ 50

Sample Input

1 2
2 2

Sample Output

1.0000
2.6667

Explanation

In the first test case, there are only 2 students. Of course they are from the same country. Hence, only 1 move is needed.

In the second test case, we wil leither take 2 moves, or three moves at most.

In the first move, the probability that we choose a pair of students from the same country, is 1/3. In this case, the game ends in 2 moves.

Otherwise, with probability 2/3, we have selected two students from different countries. Now, we will only take 2 more moves. **For the third student we ask to stand up, after the first 2 sit, we already know who his pair is from the first move.** And the remaining two will be asked to stand up in the third move.

Hence, the expected number of moves is **2 * (1/3) + 3 * (2 / 3) = 2.6667**

Author:	gamabunta
Tags	gamabunta
Date Added:	13-10-2013
Time Limit:	2 sec
Source Limit:	50000 Bytes
Languages:	C, CPP 4.8.1, JAVA

SUBMIT

Comments

Need help? Post a comment. But before that please spare a moment to read the **guidelines**.

Your name:
acmkn13tm087

Comment: *

Save

The time now is: 11:00:38 AM

CodeChef - A Platform for Aspiring Programmers

CodeChef was created as a platform to help programmers make it big in the world of algorithms, **computer programming** and **programming contests**. At CodeChef we work hard to revive the geek in you by hosting a **programming contest** at the start of the month and another smaller programming challenge in the middle of the month. We also aim to have training sessions and discussions related to **algorithms, binary search**, technicalities like **array size** and the likes. Apart from providing a platform for **programming competitions**, CodeChef also has various algorithm tutorials and forum discussions to help those who are new to the world of **computer programming**.

Practice Section - A Place to hone your 'Computer Programming Skills'

Try your hand at one of our many practice problems and submit your solution in a language of your choice. Our **programming contest** judge accepts solutions in over 35+ programming languages. Preparing for coding contests were never this much fun! Receive points, and move up through the CodeChef ranks. Use our practice section to better prepare yourself for the multiple **programming challenges** that take place through-out the month on CodeChef.

Compete - Monthly Programming Contests and Cook-offs

Here is where you can show off your **computer programming** skills. Take part in our 10 day long monthly **coding contest** and the shorter format Cook-off **coding contest**. Put yourself up for recognition and win great prizes. Our **programming contests** have prizes worth up to Rs.20,000 and $700lots more CodeChef goodies up for grabs.

Discuss

Are you new to **computer programming**? Do you need help with algorithms? Then be a part of CodeChef's Forums and interact with all our programmers - they love helping out other programmers and sharing their ideas. Have discussions around **binary search, array size, branch-and-bound, Dijkstra's algorithm, Encryption algorithm** and more by visiting the CodeChef Forums and Wiki section.

CodeChef Community

As part of our Educational initiative, we give institutes the opportunity to associate with CodeChef in the form of Campus Chapters. Hosting **online programming competitions** is not the only feature on CodeChef. You can also host a **coding contest** for your institute on CodeChef, organize an **algorithm** event and be a guest author on our blog.

Go For Gold

The Go for Gold Initiative was launched about a year after CodeChef was incepted, to help prepare Indian students for the **ACM ICPC** World Finals competition. In the run up to the **ACM ICPC** competition, the Go for Gold initiative uses CodeChef as a platform to train students for the **ACM ICPC** competition via multiple warm up contests. As an added incentive the Go for Gold initiative is also offering over Rs.8 lacs to the Indian team that beats the 29th position at the **ACM ICPC** world finals. Find out more about the Go for Gold and the **ACM ICPC** competition here.