So you want to be a 2^n-aire?

Problem ID: 2naire
CPU Time limit: 1 second
Memory limit: 1024 MB
Difficulty: 2.9

Author: Don Reble
Source: Waterloo Programming Contest 2005-09-17
License: For educational use only

The player starts with a prize of \$1, and is asked a sequence of n questions. For each question, he may

- quit and keep his prize.
- answer the question. If wrong, he quits with nothing. If correct, the prize is doubled, and he continues with the next question.

After the last question, he quits with his prize. The player wants to maximize his expected prize.

Once each question is asked, the player is able to assess the probability p that he will be able to answer it. For each question independently, we assume that p is a random variable uniformly distributed over the range $t \dots 1$.

Input

Input is a number of lines, each with two numbers: an integer $1 \leq n \leq 30$, and a real $0 \leq t \leq 1$. Input is terminated by a line containing `0 0`. This line should not be processed. There are at most 300 test cases.

Output

For each input n and t, print the player's expected prize, if he plays the best strategy. Output should be rounded to three fractional digits.

Sample Input 1

```
1 0.3
2 0.6
24 0.25
0 0
```

Sample Output 1

```
1.357
2.560
230.138
```