



ASOCIACION
COLOMBIANA
DE INGENIEROS
DE SISTEMAS



Universidad de
los Andes



XXI Maratón Nacional de Programación
ACIS REDIS 2009
ACM ICPC

Problemas Warm-up

(Este conjunto contiene 2 problemas de calentamiento; páginas numeradas de 1 a 3)

Problem A

The Birthday Paradox

Source file name: birthday.c, birthday.cpp or birthday.java

Given n , the number of participants in a birthday party, you have to estimate the probability that at least one pair of them do have the same birthday (month and day), assuming that a year has 365 days.

To estimate that probability you may think that the birthday distribution is uniform along the year, i.e. one person may have been born at any of the 365 days of the year with the same probability.

Input

Each line of the input contains an integer n ($0 \leq n \leq 365$) specifying the number of participants in the party.

The input must be read from the file birthday.in.

Output

For each case in the input, print one line with the probability that at least one pair of the n participants do have the same birthday. Round the results to six decimal places. The floating point delimiter must be '.' (i.e. the dot). The rounding applies towards the *nearest neighbor* unless both neighbors are equidistant, in which case the result is rounded up (e.g. 0.1234562 is rounded to 0.123456; 0.1234566 is rounded to 0.123457; 0.1234565 is rounded to 0.1234567, etc.).

The output must be written to standard output.

Sample input	Output for the sample input
365	1.000000
0	0.000000
23	0.507297

Problem B

Hello Kitty

Source file name: hkitty.c, hkitty.cpp or hkitty.java

And if he left off dreaming about you ... Hello, Kitty!
Through the Looking-Glass, VI (free version)

Kitty sends a kind of original email messages to her friend Garf. To write a message, she chooses a word W and a number n and replicates W n times horizontally. Then she repeats this string in the next line, but rotating the characters once to the left. And she repeats this 'rotate-and-output' process until the word W appears displayed as the first column of the rectangular pattern that she produces.

As an example, when she chooses the word Hello and the number 3, she gets the pattern:

```

HelloHelloHello
elloHelloHelloH
lloHelloHelloHe
lHelloHelloHel
oHelloHelloHell

```

Kitty has been sending such emails during the last three years. Recently, Garf told her that perhaps her work may be automatized with a software to produce Kitty's patterns. Could you help her?

Input

The input file contains several test cases, each one of them in a separate line. Each test case has a word and a positive integer that should generate the corresponding rectangular pattern. The word is a string of alphabetic characters (a..z). The number is less than 10.

A line whose contents is a single period character means the end of the input (this last line is not to be processed).

The input must be read from the file hkitty.in.

Output

Output texts for each input case are presented in the same order that input is read. For each test case the answer must be a left aligned Kitty pattern corresponding to the input.

The output must be written to standard output.

Sample input	Output for the sample input
Love 1	Love
Kitty 2	oveL
.	veLo
	eLov
	KittyKitty
	ittyKittyK
	ttyKittyKi
	tyKittyKit
	yKittyKitt