ACM International Collegiate Programming Contest
1999 East Central Regional Practice Contest
University of Waterloo
November 12, 1999

Sponsored by IBM

Rules:

1. All questions require you to read the test data from standard input and write results to standard output. You cannot use files for input or output.
   - No whitespace should appear at the end of a line, and all lines should be terminated with a new-line.
   - Tabs should never be used.
   - Output must correspond *exactly* to the provided sample output, including (mis)spelling and spacing. Multiple spaces will not be used in any of the judges' output, except where expressly stated.
2. All programs will be re-compiled prior to testing with the judges' data.
3. Non-standard libraries cannot be used in your solutions. The Standard Template Library (STL) and C++ string libraries are allowed.
4. Programming style is not considered in this contest. You are free to code in whatever style you prefer. Documentation is not required.
5. All communication with the judges will be handled by the submit command.
6. Each solution (C, C++, Pascal) is given 5 CPU seconds and 16 MB of memory to run for each input file. The run time and memory limits on Java solutions will be given separately in another document.
7. The allowed programming languages are C, C++, Java, and Pascal.
8. Judges' decisions are to be considered final. No cheating will be tolerated.
9. There are **four** questions to be completed in **eighty minutes**.

Problem A: Word Reversal

For each list of words, output a line with each word reversed without changing the order of the words.

Input

You will be given a number of test cases. The first line contains a positive integer indicating the number of cases to follow. Each case is given on a line containing a list of words separated by one space, and each word contains only uppercase and lowercase letters.

Output

For each test case, print the output on one line.

Sample Input

```
3
I am happy today
To be or not to be
I want to win the practice contest
```

Sample Output

```
I ma yppah yadot
oT eb ro ton ot eb
I tnaw ot niw eht ecitcarp tsetnoc
```

Problem B: A Mathematical Curiosity

Given two integers n and m, count the number of pairs of integers (a, b) such that $0 < a < b < n$ and $(a^2 + b^2 + m)/(ab)$ is an integer.

Input

You will be given a number of cases in the input. Each case is specified by a line containing the integers n and m. The end of input is indicated by a case in which $n = m = 0$. You may assume that $0 < n \leq 100$.

Output

For each case, print the case number as well as the number of pairs (a, b) satisfying the given property. Print the output for each case on one line in the format as shown below.

Sample Input

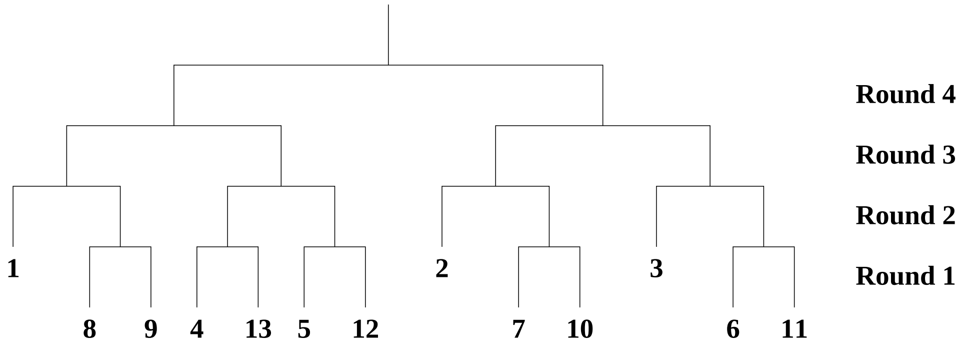
```
10 1
20 3
30 4
0 0
```

Sample Output

```
Case 1: 2
Case 2: 4
Case 3: 5
```

Problem C: Tournament Seeding

In many competitions, players are given seeding numbers to represent their relative strength, where the best player is given seed #1, the second best is given seed #2, etc. The *seeding* of a single elimination tournament is an arrangement of matches which keeps the best players from playing each other until as late as possible (the last round with players seeded 1 and 2). Define the *strength* for a match as the sum of the two seeds in the match, and the *ideal strength* as the strength for the match assuming the best two players make it to that match (or in other words, the minimum sum of the two seeds which could play in that match). The total number of rounds r needed for a tournament of n contestants is given by the formula $r = \lceil \log_2 n \rceil$. If we call the finals of the tournament round r, the semi-finals round $r - 1$, etc., then the way in which seeding is done can be described as follows: in round k, we arrange players such that the ideal match strength for all matches in that round is $2^{r-k+1} + 1$. For example, the seeding of a match of 13 people would look as follow:



You are asked to solve the following problem: given the values of n and m, determine the earliest round in a tournament of n players in which a match strength of m could occur. You may assume that the seeding is done in the manner described above.

Input

You will be given a number of cases in the input. Each case will be specified on one line, consisting of two integers n and m, where $2 \leq n \leq 100$ and $m \geq 3$. You may assume in each case that a match strength of m can occur during some round. Input is terminated by a case in which $n = m = 0$.

Output

For each test case, print the case number and the earliest round for that case in the format shown below.

Sample Input

```
100 3
13 10
0 0
```

Sample Output

```
Case 1: Round 7
Case 2: Round 2
```

Problem D: Niven Numbers

A Niven number is a number such that the sum of its digits divides itself. For example, 111 is a Niven number because the sum of its digits is 3, which divides 111. We can also specify a number in another base b, and a number in base b is a Niven number if the sum of its digits divides its value.

Given b ($2 \leq b \leq 10$) and a number in base b, determine whether it is a Niven number or not.

Input

You will be given a number of test cases. Each line of input contains the base b, followed by a string of digits representing a positive integer in that base. There are no leading zeroes. The input is terminated by a line consisting of 0 alone.

Output

For each case, print "yes" on a line if the given number is a Niven number, and "no" otherwise.

Sample Input

```
10 111
2 110
10 123
6 1000
8 2314
0
```

Sample Output

```
yes
yes
no
yes
no
```