programación creativa temporada 2010

The 2010 ACM UCLA Practice Two Programming Contest

June 12, 2010

Universidad Centroccidental Lisandro Alvarado

Contest session

This problem set contains 8 problems; pages are numbered from 1 to 9.

Problem A

Another Rose

Source file name: `rose.c`, `rose.cpp` *or* `rose.java`
You must read from standard input and write to standard output.

*"mi vorrai sposare
per tutto quello che io dolcemente ti darè?
se vorrai, rispondi solo si o no
farai così o no?"*

Ok, Rose is a girl with a black cat. A crazy boy called Noel told her: "*Marry with me*", and Rose says "*No, I cannot*", but she then decided to reconsider the offer. With all of this, Rose forgot to do her job (she works in a box factory) and the job is for this week so Rose is in problems.

Help to rose to write a little program to determinate: "*In how many ways can you choose k elements out of n elements, not taking order into account?*"

Note about what the Rose's boss says: "*The program must be really fast.*"

Input

The input is described by several lines, each one with two integers, k and n, and ends with $k = n = -1$.

Output

For each test case, you must print how many ways you can choose k elements out of n elements, not taking order into account. You can assume that number always fits in a 32-bit signed integer.

Sample input

```
4 2
49 6
-1 -1

```

Sample output

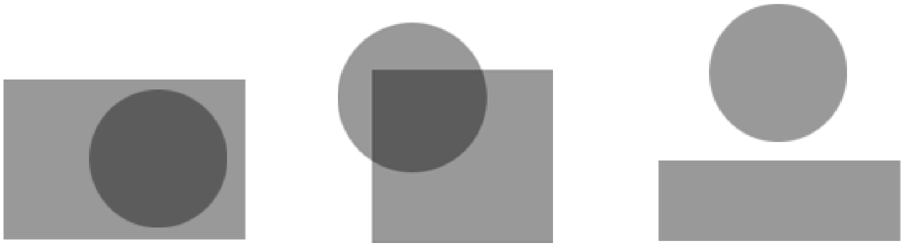
```
6
13983816

```

Problem B

Beware of Sesom

Source file name: `sesom.c`, `sesom.cpp` *or* `sesom.java`
You must read from standard input and write to standard output.

Nothing wants to be Sesom. Well, he's a crazy guy, he have a lot of problems because his mind is down. Right now, Sesom wants to hunt ducks. Why? Because he's a dumb-ass. Still, Sesom is practicing with a gun target shooting; you must help to Sesom to determinate if he did shoot the target. The target is a rectangle with a circle, not necessarily one within another, as shown in the figures below.



Input

The input contains several test cases, the first line is N, the number of cases $(2 \leq N \leq 999)$ and then N lines follows, each one with 9 integers: x_1, y_1, x_2, y_2, x_c, y_c, r, x_p, y_p. Where (x_1, y_1) are the coordinates of the upper left corner, and (x_2, y_2) is the lower right corner. (x_c, y_c) represents the circle center, and r the radius. Similarly (x_p, y_p) is the position where Sesom shot the wall.

Output

If the Sesom shot is contain in the circle, print "*Sesom hunt the Duck*", if the shot is inside of the rectangle but is outside of the circle then print "*Sesom hunt the toy Duck*", If the shot given is outside of both figures print "*Sesom is really stupid*".

Sample input

```
3
0 0 10 5 7 2 1 7 2
0 0 10 5 7 2 1 8 2
0 0 10 5 7 2 1 -3 5

```

Sample output

```
Sesom hunt the Duck
Sesom hunt the toy Duck
Sesom is really stupid

```

Problem C

Cherry Blossom Clinic

Source file name: `cherry.c`, `cherry.cpp` *or* `cherry.java`
You must read from standard input and write to standard output.

Roy likes to talk to trees, so you may guess what the clinic name stands for. However, in one of the few, but getting better, moments of clarity, he fall in love with a girl, a beautiful one I must admit. She likes the British band Nirvana (I don't think it's a valid reason to put her there but a lot of people do). Well, she's too good for him, he said to one of his woody-made friends, but still, he can't stop thinking about her.

Then he realize that she usually takes a long time thinking about how organize her eclectic British rock music records. And in an attempt to get her attention, Roy came up with an idea. He said about put a score to each album, a unique one and such one the best albums would be ranked higher. But this was something hard to do, especially if you have hundreds of albums (I have no idea about how those got to the clinic). By then, he would just rather ask about what album she prefers over another. I mean, he would ask if she like Move's Shazam more than Beatles's Revolver, and so on.

And so he did. Right now, he has a list of albums preferences, and it's his turn to get a way to order them for her love, and he's asking for your help. By the way, Roy thinks you're a smart talking tree.

Input

The input describes several test cases, in the first line there is T, the number of cases ($T \leq 100$) and then, for each one there's a list of preferences. Any list starts with N, $1 \leq N \leq 100$, the number of consulted queries and then N pair of lines follows, each describe that the album in the first line if preferred over the name showed in the second line. The names will be inside quotes and would have not more than 50 characters, and both will be separated by a space. You can assume that there won't be incoherencies.

Output

For each test case, print the case number and the list of the albums ordered by preferences, that is, any album in the list must be favored better over any of the following items. Each name must be in one line, and after the entire list, there should be a blank line. If do exists multiples solutions, then print the one who comes lexicographically first. Look at the sample output for exact format.

Sample input

```
2
8
Shazam
Revolver
Sgt. Pepper's Lonely Hearts Club Band
S.F. Sorrow
Story of Simon Simopath
In the Court of the Crimson King
Shazam
Days of Future Passed
Revolver
Electric Light Orchestra
Sgt. Pepper's Lonely Hearts Club Band
Electric Light Orchestra
In the Court of the Crimson King
S.F. Sorrow
Revolver
Sgt. Pepper's Lonely Hearts Club Band
1
The Who Sell Out
Queen II

```

Sample output

```
Case #1:
Shazam
Days of Future Passed
Revolver
Sgt. Pepper's Lonely Hearts Club Band
Electric Light Orchestra
Story of Simon Simopath
In the Court of the Crimson King
S.F. Sorrow

Case #2:
The Who Sell Out
Queen II


```

Problem D

Damn It!

Source file name: `damn.c`, `damn.cpp` *or* `damn.java`
You must read from standard input and write to standard output.

Flavio is a student of Cs. Computation of a weird country, and he have a big problem with one of his investigations, he is developing a encryption algorithm like SHA1 as homework, but Flavio doesn't know how to count a *insane bit*, but now you say: What is a insane bit? Ok, if a bit of a number is 1 and the next bit is also 1 then we can say that the number has 1 insane bit.

Examples:

Number	Binary	Insane Bits
12	1100	1
15	1111	3

In the same way, do you remember about XOR operation? Well, given two bits, the XOR operation is defined as:

```
0 XOR 0 → 0
0 XOR 1 → 1
1 XOR 0 → 1
1 XOR 1 → 0
```

The problem is that you must help to Flavio with his algorithm. Given two numbers, P and Q, you can obtain a number K, making $K = P\ \boldsymbol{XOR}\ Q$, and you must count the insane bits in the K number.

Input

You given several test cases, with pairs of positive numbers, P and Q, the input ends with an EOF marker. Each number fits in a 32-bit signed integer.

Output

For every test case, print one line of the form "*Case A: B*", without quotes, where A is the serial of output (starting from 1) and B is the number of insane bits.

Sample input

```
12 3

```

Sample output

```
Case 1: 3

```

Problem E

Exponentiation

Source file name: `exp.c`, `exp.cpp` *or* `exp.java`
You must read from standard input and write to standard output.

Modular arithmetic makes the life easier for those who get annoyed by big number problems. For example, if you're asked to get $(18{,}756 + 3{,}514) \bmod 100$, this is the same than $(18{,}756 \bmod 100 + 3{,}514 \bmod 100) \bmod 100 = (56 + 14) \bmod 100 = 70$. This can avoid you some overflow issues. I mean, try to execute the next lines.

```
int a = 2031456487, b = 500326123;
cout << (a + b) % 654 << endl;
```

And you'll realize that doesn't print the real answer, which you can get with `((a % 654) + (b % 654)) % 654` and so you don't get any overflow. This also works for multiplication operations, but it doesn't help in subtraction or division problems.

I want also you know about exponentiation by squaring. Look at this, $345^{100} = (345^{50})^2$, right?. So you calculate 345^{50} first, and then you multiply it by itself so you get 345^{100} without making a lot of multiplications. What about 311^{37}? Well, that's equal to $311 \times 311^{36} = 311 \times (311^{18})^2$, isn't?. You can generalize this as:

$$a^b = \begin{cases} 1 & \text{if } b = 0 \\ a^{b/2} \times a^{b/2} & \text{if } b \text{ is even} \\ a^{b-1} \times a & \text{if } b \text{ is odd} \end{cases}$$

And with this you can get a^b in logarithmic time, i.e., very fast. In this problem, I will ask you the last three digits of a^b. Well, to be more precise, $a^b \bmod 1000$.

Input

There're several test cases, each one includes a and b where $0 < a, b < 2^{31}$. The input ends with $a = b = 0$.

Output

For each test case print one line containing one integer $a^b \bmod 1000$.

Sample input

```
2 4
3 100
5365 123456789
0 0

```

Sample output

```
16
1
125

```

Problem F

Fibonacci

Source file name: `fib.c`, `fib.cpp` *or* `fib.java`
You must read from standard input and write to standard output.

Fibonacci sequence is given as: $0, 1, 1, 2, 3, 5, 8, 13, \ldots$ Every element (except the first two - 0 and 1) is the sum of the previous two. Formally, is defined as the next recurrence relation:

$$F_0 = 0, \;\; F_1 = 1, \;\; F_n = F_{n-1} + F_{n-2}, \;\; n \geq 2$$

There exist a lot of identities and applications related to Fibonacci sequence. There's even a magazine dedicated to only this relation who has been released for almost forty years! But by know, I only ask you to give the Fibonacci number F_n for very high values of n. Also I'll give you a hint about how you can do it in little time. Look, you can see that for any $n \geq 1$,

$$\begin{cases} F_n = F_{n-1} + F_{n-2} \\ F_{n-1} = F_{n-1} \end{cases} \Leftrightarrow \begin{bmatrix} F_n \\ F_{n-1} \end{bmatrix} = \begin{bmatrix} F_{n-1} + F_{n-2} \\ F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} F_{n-1} \\ F_{n-2} \end{bmatrix}$$

But then,

$$\begin{bmatrix} F_n \\ F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} F_{n-2} \\ F_{n-3} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^2 \begin{bmatrix} F_{n-2} \\ F_{n-3} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^3 \begin{bmatrix} F_{n-3} \\ F_{n-4} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^k \begin{bmatrix} F_{n-k} \\ F_{n-k-1} \end{bmatrix}$$

So, if we make $n - k - 1 = 0$ then we can get that,

$$\begin{bmatrix} F_n \\ F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{n-1} \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

Input

The input contains several test cases. Each test case consists of a single line with only one integer n, $0 \leq n < 2^{63}$. The end of input is indicated by $n = -1$.

Output

For each test case print one line containing a single integer with F_n (mod 10000007). If you aren't aware about modular arithmetic or exponentiation by squaring then take a look to problem E.

Sample input

```
1
6
100
-1

```

Sample output

```
1
8
1759647

```

Problem G

Girls Code

Source file name: `girls.c`, `girls.cpp` *or* `girls.java`
You must read from standard input and write to standard output.

Scott Walker-singed song shows how hard can be the understanding of a woman. But good news! It seems we have found a way to decode some things that they say. They appear to substitute every letter with another one and we also have the encoding mask that they use.

So your problem is to decode the string given, so then understand what they're saying and live happily after… I wish it was that easy.

The girls
they throw you from towers
they'll whip you with flowers
it depends on the hours

The girls
are still making dates
they still making you wait
and they say you are late

- The Girls and the Dogs

Input:

The input file consists of several cases. The first line have N, the number of test cases. And then follows N pair of lines, each one is a test case. The first line of each test case describes the codification, a string of 26 different characters. Let's say you have QWASFHPOL…, then A was encoded as Q, B as W, C as A, and so on.

The second line represents the message to decode. You can assume that the message has less than 1000 characters and has only uppercase letters, symbols and spaces.

Output:

For each test case, print one line with the message decoded.

Sample input

```
1
OQASEBTLIHJZDRFUMYGVXWCNPK
I LOVE YOU

```

Sample output

```
I HATE RAP

```

Problem H

Help

Source file name: `help.c`, `help.cpp` *or* `help.java`
You must read from standard input and write to standard output.

Help! We have several guesses and we need to distribute them in several tables. We will give you n, the number of people, and m, our preferred number of those per table. So, if some people left for a table, we want distribute them as uniformly as possible in the already formed tables. We need somebody, but not just anybody, can you help us?

Input:

Input includes several lines, each with two integers, n and m. Notice that $1 \leq n, m \leq 10^9$. Input ends with $n = m = 0$.

Output:

For each test case print one line starting with the number of groups to form. And then, for every table, write the number of people who we should put in that table. These integers should be in descending order and separated by one space. You can assume that the numbers of tables will never be more than 10000.

Sample input

```
4 8
6 2
20 8
0 0

```

Sample output

```
1 4
3 2 2 2
2 10 10

```