

ACM International Collegiate Programming Contest 2007-2008

Sponsored by IBM

Competencia Sudamericana de Programación 2007

Primera Fase Bolivia

Universidad Mayor de San Andrés

La Paz, 20 de Septiembre de 2007

Problem Set

El concurso consta de un conjunto de ocho (8) problemas en nueve (9) páginas.
Por favor, informe a alguno de los responsables si falta alguna de ellas.

Competencia Sudamericana de Programación 2007
Primera Fase Bolivia
20 de Septiembre de 2007

La Paz, Cochabamba y Santa Cruz

LISTA DE PROBLEMAS

- A The Hotel with Infinite Rooms
- B Back to High School Physics
- C An Easy Problem!
- D Diagonal
- E Almost Prime Numbers
- F Bolivian Sequences
- G Pay the Price
- H Meeting Prof. Miguel ...

Autor de los problemas: **Shahriar Manzoor**, de Bangladesh.

Coordinador del Elite Problem-setters Panel, colectivo de creadores de problemas para los concursos organizados en el juez automático de la Universidad de Valladolid.

Problem A: The Hotel with Infinite Rooms

The city of **HaluaRuti** has a strange hotel with infinite rooms. The groups that come to this hotel follow the following rules:

- a) At the same time only members of one group can rent the hotel.
- b) Each group comes in the morning of a day and leaves the hotel in the evening of another day.
- c) Another group comes in the very next morning after the previous group has left the hotel.
- d) A very important property of the incoming group is that it has one more member than its previous group unless it is the starting group. You will be given the no of members of the starting group.
- e) A group with n members stays for n days in the hotel. For example, if a group of four members comes on 1st August in the morning, it will leave the hotel on 4th August in the evening and the next group of five members will come on 5th August in the morning and stay for five days and so on.

Given the initial group size you will have to find the group size staying in the hotel on a specified day.

Input

The input contains round numbers S ($1 \leq S \leq 10000$) and D ($1 \leq D < 10^{15}$) in every line. S denotes the initial size of the group and D denotes that you will have to find the group size staying in the hotel on D -th day (starting from 1). All the input and output integers will be less than 10^{15} . A group size S means that on the first day a group of S members come to the hotel and stays for S days then comes a group of $S + 1$ members according to the previously described rules and so on.

Output

For each line of input, print on a single line the size of the group staying in the hotel on the D -th day.

Sample Input

```
1 6
3 10
3 14
```

Sample Output

```
3
5
6
```

Problem B: Back to High School Physics

A particle has initial velocity and acceleration. If its velocity after certain time is v then what will its displacement be in twice of that time?

Input

The input will contain two integers in each line. Each line makes one set of input. These two integers denote the value of v ($-100 \leq v \leq 100$) and t ($0 \leq t \leq 200$) (t means at the time the particle gains that velocity)

Output

For each line of input print a single integer in one line denoting the displacement in double of that time.

Sample Input

```
0 0
5 12
```

Sample Output

```
0
120
```

Problem C: An Easy Problem!

Have you heard the fact “*The base of every normal number system is 10*”? Of course, I am not talking about number systems like Stern Brockot Number System. This problem has nothing to do with this fact but may have some similarity.

You will be given an N based integer number R and you are given the guaranty that R is divisible by $(N - 1)$. You will have to print the smallest possible value for N . The range for N is $2 \leq N \leq 62$ and the digit symbols for 62 based number is (0..9 and A..Z and a..z). Similarly, the digit symbols for 61 based number system is 0..9 and A..Z and a..y) and so on.

Input

Each line in the input file will contain an integer (as defined in mathematics) number of any integer base (2..62). You will have to determine what is the smallest possible base of that number for the given conditions. No invalid number will be given as input. The largest size of the input file will be 32KB

Output

If number with such condition is not possible output the line ‘such number is impossible!’ For each line of input there will be only a single line of output. The output will always be in decimal number system.

Sample Input

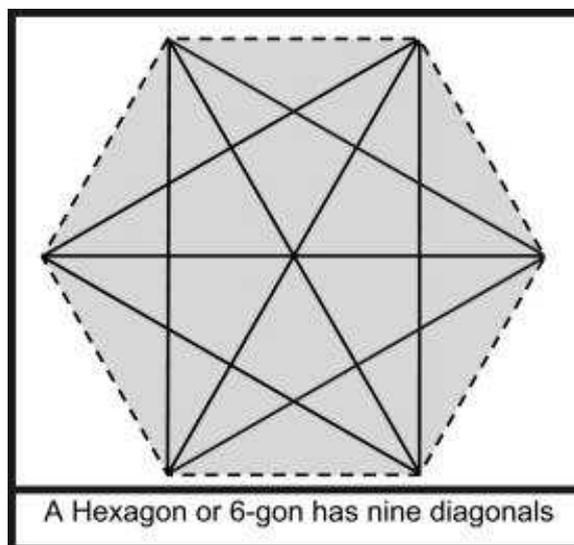
```
3
5
A
```

Sample Output

```
4
6
11
```

Problem D: Diagonal

The number of diagonals of an n -gon is not less than N . What is the minimum possible value of n ?



Input

The input file contains less than 1001 lines of inputs. Each line contains a positive integer N ($N \leq 10^{15}$) that indicates the minimum possible number of diagonals. Input is terminated by a line containing a zero. This line should not be processed.

Output

For each line of input produce one line of output, which contains the output serial number, and also the minimum possible value for n (Number of sides).

Sample Input

```
10
100
1000
0
```

Sample Output

```
Case 1: 7
Case 2: 16
Case 3: 47
```

Problem E: Almost Prime Numbers

Almost prime numbers are the non-prime numbers which are divisible by only a single prime number. In this problem your job is to write a program which finds out the number of almost prime numbers within a certain range.

Input

First line of the input file contains an integer N ($N \leq 600$) which indicates how many sets of inputs are there. Each of the next N lines make a single set of input. Each set contains two integer numbers low and $high$ ($0 < low \leq high < 10^{12}$).

Output

For each line of input except the first line you should produce one line of output. This line contains a single integer, which indicates how many almost prime numbers are within the range (inclusive) $low \dots high$.

Sample Input

```
3
1 10
1 20
1 5
```

Sample Output

```
3
4
1
```

Problem F: Bolivian Sequences

Bolivian Sequences are sequences of at most 15 letters. The allowed letters in the sequence of length 15 are A, B, C, D, E, F, G, H, I, J, K, L, M, N and O. Similarly allowed letters in a sequence of length 4 are A, B, C, D; allowed letters in a sequence of length 5 are A, B, C, D, E and so on. The first 15 letters of English alphabet have the values 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76, 77, 78 and 79 respectively. A sequence of length n ($n < 16$) formed with first n letters of English alphabet is called Bolivian Sequence if it fulfills at least one of the following conditions (Multiple fulfillment is also not harmful):

- a) If any two characters are equal. So 'ABAD' is a Bolivian sequence according to this rule only.
- b) If the absolute value of the difference of position of any two characters is equal to the absolute value of the difference of their values. So 'ABCD' is a Bolivian sequence according to this rule. Position of 'A' is 1 and position of 'B' is 2, $|2 - 1| = 1$ and $|A - B| = |65 - 66| = 1$ as well.
- c) If the value of any two adjacent characters differ by two.
- d) If the value of any two characters who has one character between them differs by 1.

You will have to find the number of Bolivian Sequences under some given constraints.

Input

The input file contains less than 502 sets of input. Each set starts with an integer n , which indicates that this set is speaking about a Bolivian sequence of length n . The next n lines define the constraints of the n positions of the sequence. A question mark means that in the corresponding position any one of the first n characters of English alphabet is feasible. If the line contains one or more alphabet instead of '?' then it means that those characters are allowed in the corresponding position. For example the second set in the sample input asks to find Bolivian Sequence of length 4, whose first character can only be 'A' and 'B' although other positions can contain any one of the characters 'A', 'B', 'C' and 'D'. Input is terminated by a set whose value of n is zero. This set should not be produced.

Output

For each set of input produce one line of output. This line should contain the output serial no as shown in the sample output and then it should contain how many Bolivian sequences are there with the given constraints and length.

Sample Input

```
4
AB
?
CD
?
0
```

Sample Output

```
Case 1: 64
```

Problem G: Pay the Price

In ancient days there was a country whose people had very interesting habits. Some of them were lazy, some were very rich, some were very poor and some were miser. Obviously, some of the rich were miser (A poor was never miser as he had little to spend) and lazy but the poor were lazy as well (As the poor were lazy they remained poor forever). The following things were true for that country

- a) As the rich were miser, no things price was more than 300 dollars (Yes! their currency was dollar).
- b) As all people were lazy, the price of everything was integer (There were no cents and so beggars always earned at least one dollar)
- c) The values of the coins were from 1 to 300 dollars, so that the rich (who were idle) could pay any price with a single coin.

Your job is to find out in how many ways one could pay a certain price using a limited number of coins (Note that the number of coins paid is limited but not the value or source. I mean there was infinite number of coins of all values). For example, by using three coins one can pay six dollars in 3 ways, $1+1+4$, $1+2+3$, and $2+2+2$. Similarly, one can pay 6 dollars using 6 coins or less in 11 ways.

Input

The input file contains several lines of input. Each line of input may contain 1, 2 or 3 integers. The first integer is always N ($0 \leq N \leq 300$), the dollar amount to be paid. All other integers are less than 1001 and non-negative.

Output

For each line of input you should output a single integer.

When there is only one integer N as input, you should output in how many ways N dollars can be paid.

When there are two integers N and $L1$ as input, then you should output in how many ways N dollars can be paid using $L1$ or less coins.

When there are three integers N , $L1$ and $L2$ as input, then you should output in how many ways N dollars can be paid using $L1, L1 + 1, \dots, L2$ coins (summing all together). Remember that $L1$ is not greater than $L2$.

Sample Input

```
6
6 3
6 2 5
6 1 6
```

Sample Output

```
11
7
9
11
```

Problem H: Meeting Prof. Miguel ...

I have always thought that someday I will meet Professor Miguel, who has allowed me to arrange so many contests. But I have managed to miss all the opportunities in reality. At last with the help of a magician I have managed to meet him in the magical **City of Hope**. The city of hope has many roads. Some of them are bi-directional and others are unidirectional. Another important property of these streets are that some of the streets are for people whose age is less than thirty and rest are for the others. This is to give the minors freedom in their activities. Each street has a certain length. Given the description of such a city and our initial positions, you will have to find the most suitable place where we can meet. The most suitable place is the place where our combined effort of reaching is minimum. You can assume that I am 25 years old and Prof. Miguel is 40+.



First meeting after five years of collaboration (Shanghai, 2005)

Input

The input contains several descriptions of cities. Each description of city is started by a integer N , which indicates how many streets are there. The next N lines contain the description of N streets.

The description of each street consists of four uppercase alphabets and an integer. The first alphabet is either 'Y' (indicates that the street is for young) or 'M' (indicates that the road is for people aged 30 or more) and the second character is either 'U' (indicates that the street is unidirectional) or 'B' (indicates that the street is bi-directional). The third and fourth characters, X and Y can be any uppercase alphabet and they indicate that place named X and Y of the city are connected (in case of unidirectional it means that there is a one-way street from X to Y) and the last non-negative integer C indicates the energy required to walk through the street. If we are in the same place we can meet each other in zero cost anyhow. Every energy value is less than 500.

After the description of the city the last line of each input contains two place names, which are the initial position of me and Prof. Miguel respectively.

A value zero for N indicates end of input.

Output

For each set of input, print the minimum energy cost and the place, which is most suitable for us to meet. If there is more than one place to meet print all of them in lexicographical order in the same line, separated by a single space. If there is no such places where we can meet then print the line 'You will never meet.'

Sample Input

```
4
Y U A B 4
Y U C A 1
M U D B 6
M B C D 2
A D
2
Y U A B 10
M U C D 20
A D
0
```

Sample Output

```
10 B
You will never meet.
```