HP CodeWars 2014

BANGALORE

Important Contest Instructions!!

Please read the following instructions carefully. They contain important information on how to package and submit your solutions to the judges. If you have any questions regarding these instructions, please ask a volunteer before the start of the contest.

Program Input/Output:

Programming tasks that require interaction with the program should prompt for user input on the screen (STDOUT) and read input from the keyboard (STDIN).

For non-interactive tasks, you will have two options for reading the input data. Using an input file named **probXX.txt** as an example, where 'XX' is the problem number.

First option, your solution may read the input from the file probXX.txt programmatically—that is, using the File I/O constructs of your programming language.

Second option is suitable for programs with large inputs by redirecting the contents of a file to your program at runtime. For example, a file named prob01.txt can be redirected to STDIN of your program using syntax like this:

```
%> java prob01 < prob01.txt
%> java –jar js.jar prob01.js < prob01.txt
%> python prob01.py < prob01.txt
%> prob01.exe < prob01.txt
```

In this example you are executing prob01 and sending the contents of the file prob01.txt to the STDIN of your program. Your program should behave exactly as it would if you were typing the input at the keyboard.

> *Tip: When entering input directly from the keyboard, type 'Ctrl-Z' <return> to signal the end of input to your program.*

All program output should always be sent to the screen (STDOUT).

Submitting your Programs

Interpreted Programs (JAVA, JavaScript, Python). Your program must be named probXX.java / probXX.js / probXX.py, where 'XX' corresponds to the problem number. Please submit only the source (.java, .js or .py). For java, the main class must be named probXX. Note the capitalization. All main and supporting classes should be in the default (or anonymous) package.

Native Programs (C. C++, etc). Your program should be named probXX.exe, where 'XX' corresponds to the problem number.

You are strongly encouraged to submit Problem #0 (on the next page) prior to the start of the competition to ensure that your build environment is compatible with the judges'.

Problem **0**

Points to Give Away

1 Points

Preface

The purpose of this question is to ensure your program can be performed correctly in the judge's computer. Strongly advise to submit this question FIRST.

This is a variation of the classic program "Hello World!"

Please print out "`Hewlett-Packard – Make it Matter`".

Input

[No input in this problem.]

Output

```
Hewlett-Packard – Make it Matter
```

Problem **1**

Positional Notation

2 Points

Question

Positional notation or place-value notation is a method of representing numbers. Positional notation is distinguished from other notations (such as Roman numerals) for its use of the same symbol for the different orders of magnitude (for example, the "units place", "tens place", "hundreds place"). This greatly simplified arithmetic led to the quick spread of the notation across the world.

Write a program to print the place value of a digit in a given number (maximum of 10 digits). If a digit repeats in the given number, find the first occurrence of the digit looking from right to left.

Input

The input consists of multiple test cases. The first line contains an integer $n < 100$ that specifies the number of test cases. The next n lines specify two numbers p and q separated by a space. The first number p is a number having a length of 10 digits or less. The second number q is a digit for which the place value needs to be found in p.

Output

Print the place value for each test case in a new line. The place value is indicated as one of the following.

- Units
- Tens
- Hundreds
- Thousands
- Ten Thousands
- Lakh
- Ten Lakhs
- Crore
- Ten Crore
- Billion

If the place value for a digit is not found, then print “`Not Found`”.

Sample Input

```
2
12345 3
55641 5
```

Sample Output

```
Hundreds
Thousands
```

Problem **2**

Arithmetic Progression

3 Points

Question

A series of number is said to be in Arithmetic Progression (AP) if the difference between any two consecutive numbers is same. For a given set of integers can you write a program to tell if they follow the Arithmetic Progression?

Input

The input consists of multiple test cases. The first line contains an integer $n < 100$ that specifies the number of test cases. The next n lines contains a set of integers separated by comma. A maximum of 50 integers can be specified in one set. The numbers in the set can range from -1000 to +1000.

Output

Print the output for each test case in a new line. For each set of numbers specify `Yes` if it follows arithmetic progression or `No` if it does not follow.

Sample Input

```
2
30,10,80,40,70,90
10,15,20,25
```

Sample Output

```
No
Yes
```

Problem **3**

Shortest Distance

3 Points

Question

A bomber airplane needs to know whether it can launch an air-to surface missile. The air to surface missiles have limited fuel and the airplane needs to know whether it can hit the target or not. Given the position of the airplane and the position of the land target, can you calculate the distance between the two?

The formula for distance between two points $P(x_1, y_1, z_1)$ and $Q(x_2, y_2, z_2)$ is given as below:

$$PQ = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

Input

The input consists of multiple test cases. The first line specifies an integer $n < 50$ number of test cases. The next n lines specifies two points between which the distance needs to be calculated. The input is provided as x_1, y_1, z_1 followed by a space and then x_2, y_2, z_2.

Output

Print out the distance between the two points with 4 decimal precision on each line.

Sample Input

```
1
76.4,12.8,4.2 77.3,11.7,0.7
```

Sample Output

```
3.7776
```

Problem 4

Combination

3 Points

Question

Combination is a way of selecting members from a grouping, such that order of selection does not matter. A combination of r from a set of n items is given as:

$$^{n}C_{r} = \frac{n!}{r! \times (n-r)!} \text{ where } r \leq n$$

If $r > n$ then $^{n}C_{r} = 0$.

Write a program that accepts positive numbers n, r and prints the maximum number of combinations that can be created.

Input

The input consists of multiple test cases. The first line specifies an integer $m < 50$ number of test cases. The next m lines consist of two positive numbers n and r separated by a space where $1 \leq n \leq 30$ and $0 \leq r \leq 30$**.**

Output

Print the maximum number of combinations for each test case in a new line.

Sample Input

```
2

5 2

10 7
```

Sample Output

```
10

120
```

Problem **5**

String Encoding

4 Points

Question

Run-length encoding (RLE) is a very simple form of data compression in which runs of data (that is, sequences in which the same data value occurs in many consecutive data elements) are stored as a single data value and count, rather than as the original run. However for data values that are not repeated the data values are retained as is.

Write a program that accepts a string and compress it using run-length encoding.

Input

The input consists of multiple test cases. The first line contains an integer $n < 100$ that specifies the number of test cases. The next n lines specify the string input that needs to be run length encoded. Maximum length of the input string can be 250 characters.

Output

Print the output of each test case in a new line. For each input string print the run length encoded string in a new line.

Sample Input

```
2
aaabbacaaaddddeefghhhh
abcdeeeefgheeeesssfffs
```

Sample Output

```
a3b2aca3d4e2fgh4
abcde4fghe4s3f3s
```

Problem **6**

Second Highest and Second Lowest

4 Points

Question

In India, many universities and schools rank their students based on the marks they get from the examinations. Given a list of marks, write a program that finds the second highest and second lowest.

Input

The input consists of multiple test cases. The first line specifies an integer $n < 50$ number of test cases. The next n lines specifies the numbers in the list of marks separated by comma. A maximum of 50 numbers can be specified in each list. The numbers in the list can range from -1000 to +1000

Output

Print the output in a new line for each test case. For each list, print the second largest number and the second smallest number separated by a space.

Sample Input

```
1
30,10,80,40,70,90
```

Sample Output

```
80 30
```

Problem **7**

Traffic Jam

5 Points

Question

Nobody can escape from the clutches of the traffic jam in Bangalore, except maybe for a lucky few. Everyone needs to wait in the signal for their turn to go ahead in a plus junction. Consider that each road in the junction leads to North, East, South and West. And for each road the signal time is varied according to the traffic policeman manning the junction. Consider that when a signal is green, vehicles in a road can clear at a rate of 36 vehicles per minute. If the incoming traffic is given, can you calculate how many vehicles are present in each road in the junction?

Input

The first line in the input specifies the number of rounds $r < 100$. The next r lines specify the incoming traffic per round, as well as the time allowed for vehicles to pass through. The input is provided as n, e, s, w and t where n, e, s and w are number of vehicles (in the range of 0 to 1000) incoming for that round. The time t represents the time in minutes that the traffic police allows the vehicles to pass through. The traffic police always allows vehicles to pass for North direction in the first round, and then the East, then the South and then the West in subsequent rounds. Note that after each round the number of traffic vehicles bundle up.

Output

The output should specify the number of vehicles in the junction for each round in a new line. Print the number of vehicles in the junction after the vehicles have passed for time t minutes in that round. The number of vehicles should be specified for North, East, South and West separated by a space.

For the sample input below, after the first round the vehicles in the North direction are allowed to pass through for 1 min. Hence the output displays as 0 after the first round for the North direction. In the second round, the vehicles are accumulated for East direction so 30 + 42 vehicles are present in East direction. The vehicles are allowed to pass through for 2 min in that direction. So 36 × 2 vehicles have passed through and the output displays 0 after the second round for East direction.

Sample Input

```
3

24 30 44 66 1

33 42 66 10 2

12 32 56 20 2
```

Sample Output

```
0 30 44 66

33 0 110 76

45 32 94 96
```

Problem **8**

Number Palindrome

5 Points

Question

A positive integer is called a palindrome if its representation in the decimal system is the same when read from left to right and from right to left. For a given positive integer k of not more than 2^{31}, write the value of the smallest palindrome larger than k to output. Numbers are always displayed without leading zeros.

Input

The first line contains integer t, the number of test cases. Integers k are given in the next t lines.

Output

For each k, output the smallest palindrome larger than k.

Sample Input

```
3
6
808
2133
```

Sample Output

```
11
818
2222
```

Problem **9**

Angle of elevation

6 Points

Question

A tanker having a height of 1m needs to fire an enemy outpost which is 6.3m tall. The enemy outposts are at various distances in the field. What should be the angle of the elevation for the gun to blow out the top of enemy outpost?

The formulae for finding the angle of elevation is given as:

$$\tan\theta = \frac{height}{distance}$$

Input

The first line specifies the number of outposts $n < 100$ in the field for which the angle of elevation needs to be found. The next n lines specifies the distance between the tanker and the enemy outpost.

Output

Print out the angle of elevation in degrees with two decimal precision.

Sample Input

```
4
15
18
20
32
```

Sample Output

```
19.45 degrees
16.40 degrees
14.83 degrees
9.40 degrees
```

Problem **10**

Fair Market Price

6 Points

Question

Based on the demand and supply of almonds, the fair market price of almonds are determined. There are s number of suppliers and r number of consumers. Each supplier produces x_i quantity of almonds and each consumer requires y_i quantity of almonds. The total quantity of almonds produced usually match the total quantity of almonds required. However when the total quantity of almonds produced is more than the total quantity of almonds required, the price of almonds drop by Rs. 3/- per quantity. And when the total quantity of almonds produced is less than the total quantity of almonds required, the price of almonds increase by Rs. 4/- per quantity.

Input

The first line specifies the integers s, r, p and n separated with spaces where $s < 20$ is the number of suppliers, $r < 20$ is the number of consumers, $p < 5000$ is the initial fair market price and $n < 100$ is the number of rounds. The next n line specifies the quantities of the almonds produced and required. The quantities are specified as $x_1, x_2, \dots x_s; y_1, y_2, \dots y_r$ where each element $x_i < 5000$ is the quantity of almond produced by each supplier and $y_i < 5000$ is the quantity of almond required by each consumer. Note that there is a semi-colon ';' between the quantities of almonds produced and required. After each round the excess of almonds either that was produced by suppliers or required by consumers is valid for the next round and needs to be fulfilled in the next round.

Output

For each round specify the fair market price of almonds.

In the below sample input, after the first round the almonds that are in excess are 8 quantities, the price of almonds is reduced by 8 x 3. Hence the output for first round is 401. In the second round, the excess of 8 quantities is added up to the quantity of almonds produced. Hence the valid quantity of almonds in second round is 96. And in the second round, the total quantity of required almonds is 98 and hence there is a deficit of 2 quantities of almonds and the price of almonds increase by 2 x 4. This deficit is added up with the quantity of almonds required in third round.

Sample Input

```
3 4 425 3

21,34,12;4,50,2,3

11,45,32;10,36,40,12

10,5,7;6,0,6,8
```

Sample Output

```
401

409

409
```

Problem **11**

Multiplicative Digital Root and Persistence

6 Points

Question

There are many properties of numbers that one can investigate. The ancient Greeks were fascinated with the properties of integers. One such property is multiplicative persistence and multiplicative digital root.

Multiply all the digits of a number n by each other, repeating with the product until a single digit is obtained. The number of steps required is known as the multiplicative persistence and the final digit obtained is called the multiplicative digital root of n.

For example, the multiplicative digital root for 128 is obtained as

1 × 2 × 8 → 16 → 1 × 6 → 6.

Since the number of step taken is 2, the multiplicative persistence is 2.

Write a program to find the multiplicative persistence and multiplicative digital root of a given input number. The input number is less than 2^{32}.

Input

The input consists of multiple test cases. The first line contains an integer $m < 100$ that specifies the number of test cases. For each number specified in the next m lines find the multiplicative persistence and multiplicative digital root.

Output

For each test case, print out the multiplicative digital root and multiplicative persistence separated by a space in a new line.

Sample Input

```
2
128
6
```

Sample Output

```
6 2
6 0
```

Problem **12**

Bus Fare

8 Points

Question

A difficult task for any new conductor in BMTC buses is to calculate the exact fare for passengers requesting their destination drop points. The BMTC bus fare is calculated in stages. The charge for each stage is determined by the type of the bus. There can be multiple bus stops within a stage. If the user provides the list of bus stop names and which bus stop initiates a new stage. Calculate the charge for a passenger between two bus stops.

Consider A, B, C, D, E and F as bus stops where A and D are stages. The following are some of the conditions for fare calculation.

- The first and last stop are always considered as a stage.
- If a person boards on A and de-boards on A or before the next stage D, the fare is calculated for one stage.
- If a person boards on A or B and de-boards on next stage D, the fare is calculated for two stages.
- If a person boards on B and de-boards on E or next stage F, the fare is calculated for three stages.

Input

The first line specifies two integers r and p where $r < 50$ and $p < 15$ separated by a space. The number of stops in the ride is specified by r and the price per stage is specified by p. The next r lines specify the bus stop name from the start of the bus ride to the end. If a '`***`' is specified after a bus stop name, then it indicates that the bus stop is a stage. This forms the database for the bus ride.

The next line specifies the number of people travelling in the bus (an integer $n < 100$). The next n lines specifies the bus stop names for fare calculation. The boarding bus stop name and de-boarding bus stop name are separated by a hyphen. If a bus stop name is not present in the database then print as "`Not Valid`" for that fare calculation.

Output

Print the bus fare for each person in a new line.

Sample Input

```
6 8
KBS (Majestic)
Corporation
Shanthinagar Bus Stand ***
Krupanidhi College
Madivala ***
Electronic City
2
KBS (Majestic) - Shanthinagar Bus Stand
Corporation - Krupanidhi College
```

Sample Output

```
16
24
```

Problem **13**

Perfect Numbers

8 Points

Question

Pythagoras saw perfection in any whole number that is equal to the sum of all its divisors, except the number itself. A perfect number is a positive integer that is equal to the sum of its proper positive divisors excluding the number itself.

Write a program that takes a positive integer n and print the "Perfect Numbers" between 1 to n.

Input

The input consists of multiple test cases. The first line contains an integer $m < 100$ that specifies the number of test cases. The next m lines specify a positive integer $n < 10{,}000$ in a new line.

Output

For each test case, print all the perfect numbers separated by a comma in a new line.

Sample Input

```
2
50
10
```

Sample Output

```
6,28
6
```

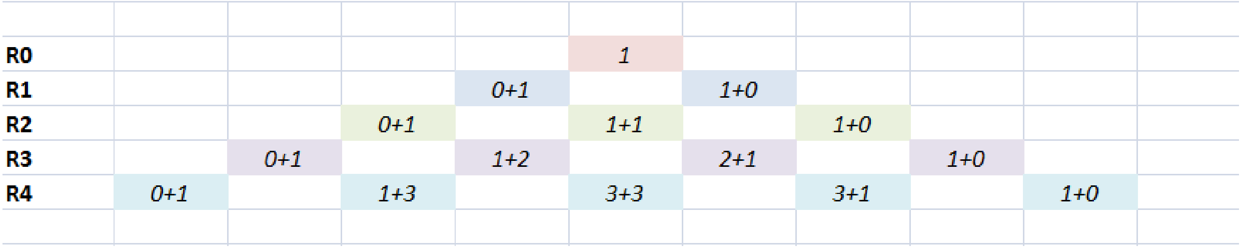
Problem 14

Pascal's Triangle

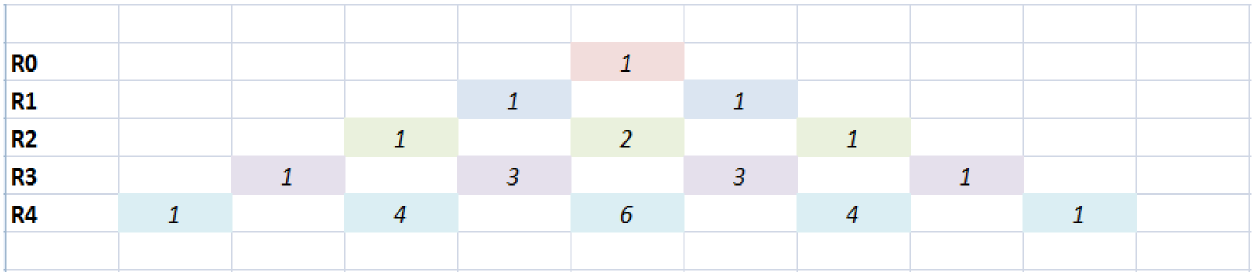
8 Points

Question

Pascal's triangle is a set of R rows, where each element is a binomial coefficient. A simple construction of the triangle proceeds in the following manner. On row 0, write only the number 1. Then, to construct the elements of following rows, add the number above and to the left with the number above and to the right to find the new value. If either the number to the right or left is not present, substitute a zero in its place. For example, the first number in the first row is 1 (the sum of 0 and 1), whereas the numbers 1 and 2 in the second row are added to produce the number 3 in the third row. Write a program that takes a positive row number and print out the elements of that row separated by a space.



R0					*1*					
R1				*0+1*		*1+0*				
R2			*0+1*		*1+1*		*1+0*			
R3		*0+1*		*1+2*		*2+1*		*1+0*		
R4	*0+1*		*1+3*		*3+3*		*3+1*		*1+0*	



R0					*1*					
R1				*1*		*1*				
R2			*1*		*2*		*1*			
R3		*1*		*3*		*3*		*1*		
R4	*1*		*4*		*6*		*4*		*1*	

Input

The input consists of multiple test cases. The first line contains an integer $m < 100$ that specifies the number of test cases. The next m lines specify a positive integer R between 0 and 30 (both inclusive) for which the elements of the row needs to be found out.

Output

For each test case, print out the elements of the row separated by a space in a new line.

Sample Input

```
1

5
```

Sample Output

```
1 5 10 10 5 1
```

Problem **15**

Word Search

9 Points

Question

One of the pass time for kids to engage themselves is word search puzzles. Given a $M \times N$ matrix of letters and a set of words, can you write a program to find the location of the given words in the matrix? These words can be formed diagonally, horizontally or vertically in any direction.

Input

The first line consists of two integers m, n where $m \le 12$ and $n \le 12$ separated by a space that specifies the size of the input matrix. The next m lines consists of n letters separated by a space. After m number of lines, an integer w is specified where $w \le 12$, that tells how many words need to be found in the matrix. The next w lines consists of words that needs to be found.

Output

Print the index of the row and column separated by a comma in parentheses for each letter. Each tuple of row and column is separated with a space for the word. Print “`Not Found`” if the word is not found in the matrix.

Sample Input

```
3 3
A G C
F A A
Z B T
3
BAG
CAT
DOG
```

Sample Output

```
(3,2) (2,2) (1,2)
(1,3) (2,3) (3,3)
Not Found
```

Problem **16**

Weights and Boxes

10 Points

Question

There are a total of 10 Boxes in which 2 boxes can accommodate 10 Kg each, 2 boxes of 20 Kg each, 2 boxes of 30 Kg each, 2 boxes of 50 Kg each and 2 boxes of 100 Kg each.

Write a program that takes weight N as input and provides the lowest possible number of boxes that can be used to accommodate the given weight in an optimal way.

Input

The program takes a single input that is the weight N < 1000.

Output

If you are able to accommodate the weights in the boxes print out the total number of boxes required in the first line. In the subsequent lines, print the number of boxes and the capacity of the boxes separated by a space. Print the highest size of the box used first and then the lowest size of the box next. If the provided weight cannot be fit into the given boxes, then print "`Not Possible`".

Sample Input 1

```
200
```

Sample Output 1

```
2

2 100
```

Sample Input 2

```
235
```

Sample Output 2

```
3

2 100

1 50
```

Problem **17**

String Pattern Matching

10 Points

Question

Write a program that takes two strings S1 and S2 as input and finds if they match according to the following definition. The first string S1 contains wild card characters and the second string S2 is a normal string. The following are allowed wild card characters in first string S1.

*	Matches with 0 or more instances of any character or set of characters.
?	Matches with any one character.

Find if the pattern given in S1 matches with the string S2

Input

The first line specifies an integer n, number of lines for which the pattern matching needs to be done. The next n lines specifies two strings in each line separated by a space. The first string S1 is the pattern string and the second string S2 is the string for which the test needs to be run.

Output

Print `Yes` if the second string S2 matches the pattern specified by S1, otherwise print `No`.

Sample Input

```
3
H*t Hewlett
Pa?k* Packard
Co*y? Company
```

Sample Output

```
Yes
Yes
No
```

Problem **18**

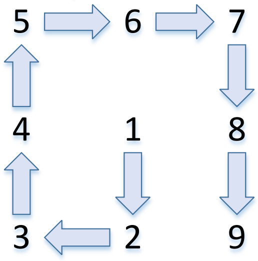
Spiral Square Matrix

12 Points

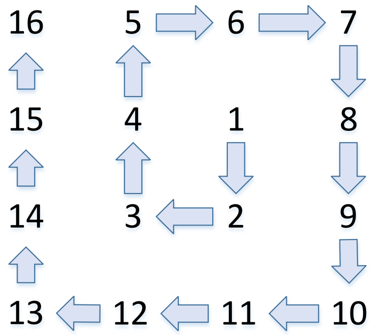
Question

Given a positive integer (M) write a program that will print the Spiral Square Matrix which has M rows and M column as shown below.

3x3 Spiral Square Matrix



4x4 Spiral Square Matrix



Input

The program takes a single input positive integer M < 50

Output

A spiral square matrix of size M x M wherein each integer is separated by a space.

Sample Input

```
3
```

Sample Output

```
5  6  7
4  1  8
3  2  9
```

Problem **19**

Morse code

12 Points

Question

Before the digital age, the most common "binary" code for radio communication was the Morse code. In Morse code, symbols are encoded as sequences of short and long pulses (called dots and dashes respectively). The following table reproduces the Morse code for the alphabet, where dots and dashes are represented as ASCII characters "." and "-":

`A .-`	`B -...`	`C -.-.`	`D -..`	`E .`	`F ..-.`
`G --.`	`H ....`	`I ..`	`J .---`	`K -.-`	`L .-..`
`M --`	`N -.`	`O ---`	`P .--.`	`Q --.-`	`R .-.`
`S ...`	`T -`	`U ..-`	`V ...-`	`W .--`	`X -..-`
`Y -.--`	`Z --..`				

Notice that in the absence of pauses between letters there might be multiple interpretations of a Morse sequence. For example, the sequence `-.-..--` could be decoded both as `CAT` or `NXT` (among others). A human Morse operator would use other context information (such as a language dictionary) to decide the appropriate decoding. But even provided with such dictionary one can obtain multiple phrases from a single Morse sequence.

Write a program that

- Reads a Morse sequence and a list of words (a dictionary)
- Find out the phrase that can be obtained from the given Morse sequence using words from the dictionary
- Print all the matching phrases for the given Morse sequence from the dictionary

Notice that we are interested in full matches, i.e. the complete Morse sequence must be matched to words in the dictionary.

Input

The first line specifies an integer $n < 20$ the number of Morse sequences to be decoded. The next n lines specifies the Morse sequence to be decoded. The next line specifies an integer m < 100 the number of words in the dictionary. The next m lines specify the words in the dictionary.

Output

Print the words that are decoded from the Morse sequence in each new line. If a Morse sequence is not found in the dictionary print as “`Not Found`”.

Sample input

```
1
.---.--.-.-.-.-
6
AT
TACK
TICK
ATTACK
DAWN
DUSK
```

Sample output

```
ATTACK
```

Problem **20**

Binary Strings

12 Points

Question

Given a positive integer n, count all possible distinct binary strings of length n such that there are no consecutive 1's. For $n = 4$ the binary strings that do not have consecutive 1's are

0000, 0001, 0010, 0100, 1000,1010, 0101 and 1001.

So the total number of binary strings for $n = 4$ is 8.

Input

The input consists of multiple test cases. The first line contains an integer $m < 100$ that specifies the number of test cases. The next m lines specify an integer $n < 100$ for which the number of binary strings that do not have consecutive 1's need to found.

Output

For each test case, print the number of binary strings that do not have consecutive 1's in a new line.

Sample Input

```
2
2
4
```

Sample Output

```
3
8
```

Problem **21**

Labyrinth

14 Points

Question

When you are on a trek with your friends you discover a labyrinth that consists of a secret treasure. The labyrinth has many rooms and in one of the room is the treasure. The rooms are connected to one another in an assorted manner. Luckily you find a map of the labyrinth consisting of the rooms and their connectivity. In order to search the rooms for the treasure, can you write a program to find the least hops between any given two rooms?

Input

The first line specifies m and n separated by spaces where m is the number of rooms in the labyrinth and n is the connectivity between rooms in the labyrinth where $m < 100$ and $n < 4000$. The next n lines specify the connectivity between two rooms as m_x, m_y separated by a comma. This forms the map of the labyrinth.

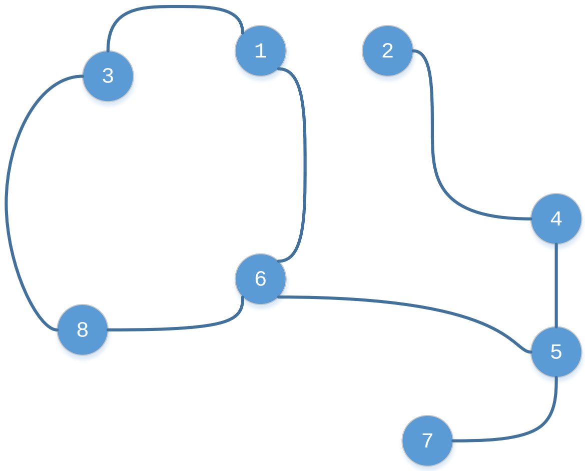
The next line, specifies an integer $p < 100$, the number of test cases. The next p lines specify two rooms p_x, p_y for which the least hops between them need to be found.

Output

Print the least number of hops between two rooms in a new line. If there is no path between two rooms, then print "`No Path Found`".

Sample Input

```
8 8
1,3
4,5
2,4
5,4
1,6
5,6
5,7
6,8
3
1,5
2,4
3,5
```



Sample Output

```
2
1
3
```