Problem A. Fail or Success?

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

Sarah is a really busy student. She usually takes so many credits every semester and at end cannot handle the pressure so doesn't do well as she supposed to do in her exams and assignments. Therefore this time she decided to calculate average hours she can study a week based on different factors like hanging out with friends, workout times and etc. and make sure the hours need to study each week don't exceed her average free hours per week. Your task is based on her decision on adding/dropping a course and the course's study hours would she have a successful or failure semester. She might add and drop a course several times during the semester.

Input

The input consists of several test cases. Each test case describes a set of courses and gives a sequence of taking/drop operations for these courses. The first line of each test case contains three integers n, m and c, where n is the number of avaibale courses ($n \leq 20$), m the number of operations performed on these courses and c is the capacity of Sarah to study per week (in hours). The following n lines contain one positive integer c_i each, the study time need for the course I (in hours). This is followed by m lines also containing one integer each, between 1 and n inclusive. They describe a sequence of taking/drop operations performed on the course. For every number, the state of that particular course is toggled, i.e. if it is currently taking, it is droped, and if it is currently not taken, it will be taken. At the beginning no course is taken. The input will be terminated by a test case starting with $n = m = c = 0$. This test case should not be processed.

Output

For each test case, first output the number of the test case. Then output whether the semester was a fail during the operation sequence. The sameser will be a fail if the sum of the hours for taking courses exceeds Sarah's capacity for studying per week. If the semster is not a fail, output the maximal hours of study per week by courses that occurred during the sequence. Output a blank line after each test case.

Example

standard input	standard output

`2 2 10`
`5`
`7`
`1`
`2`
`3 6 10`
`2`
`5`
`7`
`2`
`1`
`2`
`3`
`1`
`3`
`0 0 0`
	`Semester 1 was a fail.`
	`Semester 2 was a success.`
	`Maximal hours was 9.`

Problem B. Simple Recursion

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

Technically, a recursive function is a function that makes a call to itself. To prevent infinite recursion, you need an if-else statement (of some sort) where one branch makes a recursive call, and the other branch does not. The branch without a recursive call is usually the base case (base cases do not make recursive calls to the function). Your task is to output the evaluation of the following function :

IF $N \geq 10$, then $f(N) = f(f(N-11))$.

IF $N \leq 9$, then $f(N) = N$.

Write a program that computes $f(N)$.

Input

Each line represent a test case represented by a single integer n where $1 \leq N \leq 10^6$. The end of input is indicated by a value 0 which should not be processed. The number of test case could be as large as $250,000$ test cases.

Output

The program shall computes the value of f(N) and print it in the following format: f(N) = y where y is the computed value of f(N) .

Example

standard input	standard output

`156`
`1007`
`0`
	`f(156) = 2`
	`f(1007) = 6`

Problem C. Summation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

Given a number N. Find the sum of all the integers between 1 and N. That means you have to find the value of $1 + 2 + 3 + ... + N$.

Input

The consist of one line. A single Intger number N, $1 \leq N \leq 10^6$

Output

sum of all integers between 1 and N.

Example

standard input	standard output
5	15

Problem D. Rotating Twins

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	64 megabytes

It is not the case that twins are always exactly the same, there exists tiny differences in most cases and in some rare case either they are perfect or even rivals. Now we will go on a journey to figure out the twin status between two squares of alphabets.

A square of alphabets contains characters A-Z in each of its cell. Now if the two given squares are exactly similar in their original state then they are considered perfect. But if not, then we will try to rotate either of them and check if its a match or not, we will keep on rotating to find a match. If we find a match then the minimum number of rotation required to reach there is the status of their twin hood.

But if there is no such match between them, then it is considered that they are not related.

Input

The first input is an integer n ($1 \leq n \leq 100$), the size of the first square. Following this is a $n * n$ grid of uppercase alphabets.

The next input is an integer m ($1 \leq m \leq 100$), the size of the second square. Following this is a $m * m$ grid of uppercase alphabets.

Output

If no rotation is needed print "Perfect Twins", if 1 rotation needed print "Normal Twins", if 2 or more rotation needed print "Rival Twins" and lastly if they are not related then print "Not Related", without the quotes.

Examples

standard input	standard output
2 AB CD 2 AB CD	Perfect Twins
2 AA AB 2 AB AA	Normal Twins
2 AA BB 2 BB AA	Rival Twins
2 AB CD 2 WX YZ	Not Related

Problem E. FizzBuzz V3

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	64 megabytes

The FizzBuzz problem is a problem given by interviewer to filter out programmers who can't program. The question is simple, list down integers from 1 to 100, but if the number is divisible by 3, print "Fizz" instead. If the number is divisible by 5, print "Buzz" instead. If the number is divisible by both 3 and 5, print "FizzBuzz".

If you are like me, then you must feel some serious itch with this question. Why must it be from 1 to 100 only? Why must it be 3 and 5? Why must it be "Fizz" and "Buzz". Why only two integer? It feels so 'hardcoded' and not future proof. In actual development, it is normal and a good idea to just hardcode such simple problem rather than making a complex and error prone solution. However, this is not actual development.

Your task is to make a future proof solution for the FizzBuzz problem. Instead of from 1 to 100, you are to print from a to b (which a can be higher than b). And instead of only 3 and 5 and "Fizz" and "Buzz" , you are given n integers d_i and n string s_i.

Your task is to print integers, each in its own line, from a to b. If the integer is divisible by the integer d_i, print s_i. If the integer is divisible by more than one d_i (where $1 \leq i \leq n$) print all the corresponding e_i with no space between them, in order which they are inputted. There can be e_i which is equal to e_j where $i \neq j$.

Input

The first line consist of three integer a, b, n $(1 \leq a, b \leq 10^5)$ $(1 \leq n \leq 100)$.

The next n line consist of an integer and a string d_i, s_i $(1 \leq d_i \leq 10^5)$. s_i does not contain any spaces. s_i will not exceed 10 character.

Output

Print integers from a to b in its own line.

If it is divisible by any integer d_i $(1 \leq i \leq n)$, then print the corresponding s_i.

If it is divisible by more than one d_i, print all s_i in order given in the input without space between them. (See example 2).

Examples

standard input	standard output
9 16 2 3 Fizz 5 Buzz	Fizz Buzz 11 Fizz 13 14 FizzBuzz 16
20 10 4 3 Ketigaan 2 Keduaan 5 Kelimaan 2 DuaLagi	KeduaanKelimaanDuaLagi 19 KetigaanKeduaanDuaLagi 17 KeduaanDuaLagi KetigaanKelimaan KeduaanDuaLagi 13 KetigaanKeduaanDuaLagi 11 KeduaanKelimaanDuaLagi

Note

In example two, notice how the divisor 2 is repeated and the order it is given in the input matters in the output.