For All problems

Memory Limit : 64 MB
Time Limit: 2sc

Problem A: Pizza

Traditionally after the Local Contest, judges and contestants go to their favourite restaurant. The contestants are really hungry after trying hard for five hours. To get their pizza as quickly as possible, they just decided to order one big pizza for all instead of several small ones. They wonder whether it is possible to put the big rectangular pizza on the surface of the round table such that it does not overhang the border of the table. Write a program that helps them!

Input Specification

Input starts with an integer number r, the radius of the surface of the round table the contestants are sitting at, $1 \leq r \leq 1000$. Then follow 2 integer numbers w and l specifying the width and the length of the pizza, $1 \leq w \leq l \leq 1000$.

Output Specification

Output whether the ordered pizza will fit on the table or not. Adhere to the format shown in the sample output. A pizza which just touches the border of the table without intersecting it is considered fitting on the table, see example 3 for clarification.

Sample Input	Sample Output
38 40 60	Pizza fits on the table
35 20 70	Pizza does not fit on the table
50 60 80	Pizza fits on the table

Problem B: Hard to Believe, but True!

The fight goes on, whether to store numbers starting with their most significant digit or their least significant digit. Sometimes this is also called the "Endian War". The battleground dates far back into the early days of computer science. Joe Stoy, in his (by the way excellent) book "Denotational Semantics", tells following story:

"The decision which way round the digits run is, of course, mathematically trivial. Indeed, one early British computer had numbers running from right to left (because the spot on an oscilloscope tube runs from left to right, but in serial logic the least significant digits are dealt with first). Turing used to mystify audiences at public lectures when, quite by accident, he would slip into this mode even for decimal arithmetic, and write things like 73+42=16. The next version of the machine was made more conventional simply by crossing the *x*-deflection wires: this, however, worried the engineers, whose waveforms were all backwards. That problem was in turn solved by providing a little window so that the engineers (who tended to be behind the computer anyway) could view the oscilloscope screen from the back.
[C. Strachey - private communication.]"
You will play the role of the audience and judge on the truth value of Turing's equations.

Input Specification

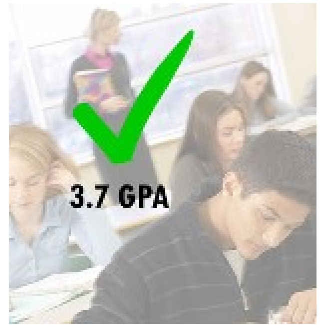
Input specifies on a single line a Turing equation. A Turing equation has the form "*a+b=c*", where *a, b, c* are numbers made up of the digits *0,...,9*. Each number will consist of at most *7* digits. This includes possible leading or trailing zeros. The equations will not contain any spaces.

Output Specification

Generate a line containing the word "True" or the word "False", if the equation is true or false, respectively, in Turing's interpretation, i.e. the numbers being read backwards.

Sample Input	Sample Output
73+42=16	True
5+8=13	False
10+20=30	True
0001000+000200=00030	False
1234+5=1239	False
1+0=0	True
7000+8000=51	True

Problem C: GPA



During the last years rating system in RAU was completely changed. Instead of getting marks 2,3,4,5 now marks are numbers [0,100].

Students of non mathematical faculties can't count their own GPA (Grade Point Average). And You must help them, and count their GPA.

During the year students studies N subjects. And have M modules from each subject, and one exam. Some modules are more important, that why each module has coefficient, a number between 0 and 1 inclusive. Sum of all coefficients for one subject is equal to 1.

For each subject You must first find the Module Average, than Average Mark. After this You must find Average Mark Between All Subjects, and then GPA.

Let students mark for j-th module of i-th subject is $A_{i,j}$, $0 \le A_{i,j} \le 100$, $1 \le i \le n$, $1 \le j \le m$

Coefficients for j-th module of i-th subject is $B_{i,j}$. $B_{i,1}+B_{i,2}+..+B_{i,m} = 1$, $0 \le B_{i,j} \le 1$, $1 \le i \le n$, $1 \le j \le m$

Coefficient between Module Average and exam for i-th subject is P_i , $0 \le P_i \le 1$, $1 \le i \le n$

Students' exam mark for i-th subject is E_i. $0 \le E_i \le 100$, $1 \le i \le n$

Coefficient of important of i-th subject is T_i. $T_1+T_2+..+T_n = 1$, $0 \le T_i \le 1$, $1 \le i \le n$

For i-th subject $(\text{Module Average})_i = B_{i,1}*A_{i,1} + B_{i,2} * A_{i,2} + .. B_{i,m}*A_{i,m}$.

For i-th subject $(\text{Average mark})_i = (\text{Module Average})_i*P_i + E_i*(1-P_i)$

Average Mark Between All Subjects = $(\text{Average mark})_1*T_1+(\text{Average mark})_2*T_2 + ... + (\text{Average mark})_n*T_n$

GPA = (Average Mark Between All Subjects) / 25

For example let n = 4, m =2 and

$A_{i,j}$

8	76
77	40
100	100
80	96

$B_{i,j}$

0.4	0.6
0.5	0.5
0.5	0.5
0.3	0.7

P_i

0.3
0.3
0.4
0.4

E_i

100
88
100
90

T_i

0.2
0.2
0.4
0.2

Module Average	Average Mark
8*0.4 + 76*0.6 = 48.8	**48.8*0.3 + 100*0.7 = 84.64**
77*0.5 + 40*0.5 = 58.5	**58.5*0.3 + 88*0.7 = 79.15**
100*0.5 + 100*0.5 = 100	**100*0.4 + 100*0.6 = 100**
80*0.3 + 96*0.7 = 91.2	**91.2*0.4 + 90*0.6 = 90.48**

Average Mark Between All Subjects = 84.64*0.2 + 79.15*0.2 + 100*0.4 +90.48*0.2 = 90.854
GPA = 90.854 / 25 = 3.63416

Input Specification

First line of input contains 2 numbers n and m. $0<n,m\leq 10$. Than follows specification of $A_{i,j}$ on n lines, each contains m integer numbers. Than follows specification of $B_{i,j}$ on n lines, each contains m numbers. Than follows n numbers of P_i Than n numbers of E_i . Than follows n numbers of T_i .

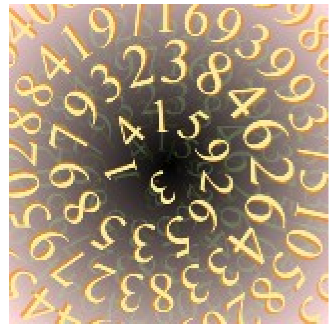
Output Specification

Output only one number GPA, with 5 digits after decimal point.

Sample Input	Sample Output
4 2 8 76 77 40 100 100 80 96 0.4 0.6 0.5 0.5 0.5 0.5 0.3 0.7 0.3 0.3 0.4 0.4 100 88 100 90 0.2 0.2 0.4 0.2	**3.63416**

Author: Eduard Piliposyan

Problem D: Square



Multiplying very large numbers is not easy. For example finding square of some number which has more than 10^6 digits is vary hard.

Now you must find square of very big numbers. But numbers for this problem are very interesting. They have this form **P00…0P** , P is a digit $1 \le P \le 9$ and number of 0s is N. For example when P = 1, and N = 3 we get 10001, P = 5 , N = 1 then 505, p=9, N=0 then 99.

Problem is to find the square of such numbers. For example

$(505)^2 = 255025$

$(99)^2 = 9801$

$(1001)^2 = 1002001$

Input Specification

Input contains 2 numbers P and N , $1 \le P \le 9$ $0 \le N \le 10^5$

Output Specification

Output one number, the square of given number.

Sample Input	Sample Output
5 1	255025
9 0	9801
1 2	1002001

Author: Eduard Piliposyan

Problem E: Find Number

For given number P, find the smallest N digit number, such that sum of digits of that number is equal to P.

Input Specifications

Input contains two number P and N , $0 \le P \le 10^5$ $1 \le N \le 10000$.

Output Specifications

Output -1 if there is no such N digit number, and output the smallest such number, otherwise. *This number must not contain leading 0s, for example you can't write 015 .*

Sample Input	Sample Output
15 3	159
30 2	-1
2 5	10001

Author: Eduard Piliposyan

Problem F: A concrete simulation

0	3	2	5	4	7	6	9	8
3	0	1	2	3	4	5	6	7
2	1	0	3	2	5	4	7	6
5	2	3	0	1	2	3	4	5
4	3	2	1	0	3	2	5	4
7	4	5	2	3	0	1	2	3
6	5	4	3	2	1	0	3	2
9	6	7	4	5	2	3	0	1
8	7	6	5	4	3	2	1	0

You are given a matrix M of type 1234x5678. It is initially filled with integers 1...1234x5678 in row major order. Your task is to process a list of commands manipulating M. There are 4 types of commands:

"R x y" swap the x-th and y-th row of M ;
"C x y" swap the x-th and y-th column of M ;
"Q x y" write out M(x,y) ;
"W z" write out x and y where z=M(x,y).

Input Specification

A list of valid commands. Input terminated by EOF. Number of commands ≤ **100000**

Output Specification

For each "Q x y" write out one line with the current value of M(x,y), for each "W z" write out one line with the value of x and y (interpreted as above) separated by a space.

Sample Input	Sample Output
R 1 2	5679
Q 1 1	1
Q 2 1	2 1
W 1	1 1
W 5679	5680
C 1 2	2
Q 1 1	2 2
Q 2 1	1 2
W 1	
W 5679	

Problem G: Mixtures

Harry Potter has n mixtures in front of him, arranged in a row. Each mixture has one of 100 different colors (colors have numbers from 0 to 99).

He wants to mix all these mixtures together. At each step, he is going to take two mixtures that stand next to each other and mix them together, and put the resulting mixture in their place.

When mixing two mixtures of colors a and b, the resulting mixture will have the color (a+b) mod 100.

Also, there will be some smoke in the process. The amount of smoke generated when mixing two mixtures of colors a and b is a*b.

Find out what is the minimum amount of smoke that Harry can get when mixing all the mixtures together.

Input Specification

The first line of input will contain n, the number of mixtures, 1 <= n <= 100. The second line will contain n integers between 0 and 99 - the initial colors of the mixtures.

Output Specification

For each test case, output the minimum amount of smoke.

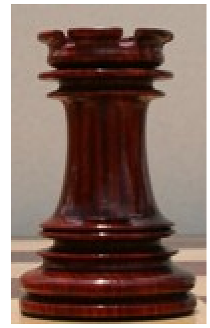
Sample Input	Sample Output
2 18 19	342
3 40 60 20	2400

In the second test case, there are two possibilities:

- first mix 40 and 60 (smoke: 2400), getting 0, then mix 0 and 20 (smoke: 0); total amount of smoke is 2400
- first mix 60 and 20 (smoke: 1200), getting 80, then mix 40 and 80 (smoke: 3200); total amount of smoke is 4400

The first scenario is a much better way to proceed.

Problem H: Rooks



You are given NxN chessboard ($1 \leq N \leq 15$), you are to determine in how many ways it is possible to place N chess rooks on that chessboard such that no two rooks share the same cell, and all cells are attacked by some rook, i. e. for each cell there are at least one rook on the same row or same column.

Input Specification

Input consists of single integer number N.

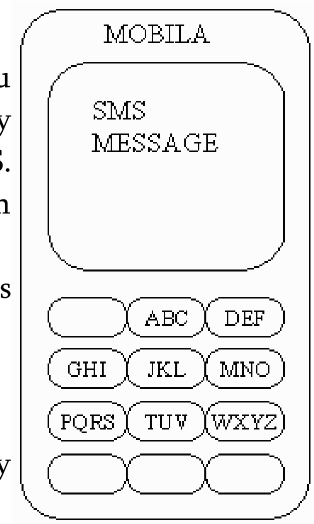
Output Specification

You should output the desired number of ways.

Sample Input	Sample Output
2	6
3	48

Author : Aram Shatakhtsyan

Problem I: SMS



You like writing SMS, you write SMS too much. Now you have find some of your old SMSs and whant to know, how many times you have to push mobile keys for writing that SMS. Remember that for writing for example letter C, you must push second key 3 times.

For example if text is HELLO then number of key presses is 2 + 2 + 3 + 3 + 3 = 12.

Input Specification

Input contains SMS on one line. SMS is constructed only using capital english letters and contains no more than 100 letters.

Output Specification

Output only one integer, number of key presses.

Sample Input	Sample Output
HELLO	13
ABCDEFGA	14

Author: Eduard Piliposyan ,
Aram Shatakhtsyan

Problem J: Binary reload

Suddenly you have found a piece of paper with a magic sequence of 0s and 1s on it, with length of N ($1 \le N \le 18$). You have figured out that you must found all such sequences, which consists of 0s and 1s with length N, and if in magic sequence on ith position there is 1, then on ith position of each your sequences must be 1. On all other positions there can be either 0 or 1. Print them in lexicographic order.

Input Specification

Input consists of one sequence of 0s and 1s with length N without spaces.

Output Specificatino

You should print desired sequences.

Sample Input	Sample Output
01101	01101 01111 11101 11111

Author: Aram Shatakhtsyan