South African Programming Olympiad
Day 1

Overview

Problem	**treasure**	**wordmath**	**xword**
Source	treasure.java treasure.py treasure.c treasure.cpp	wordmath.java wordmath.py wordmath.c wordmath.cpp	xword.java xword.py xword.c xword.cpp
Input file	stdin	stdin	stdin
Output file	stdout	stdout	stdout
Time limit	2 seconds. Python: 20 seconds.	1 second. Python: 10 seconds.	2 seconds. Python: 20 seconds.
Memory limit	256MiB	256MiB	256MiB
Detailed feedback	20	No	No
Total points	**100**	**100**	**100**

The maximum total score is 300 points.

http://olympiad2.cs.uct.ac.za/contest.html

Treasure Distribution

Introduction

The Swashbuckling Arrrrring Pirate Organisation (SAPO) has plundered a Spanish treasure galleon and stolen S silver coins. They are now dividing up the coins between the P pirates. The pirates are ordered by seniority from 1 (the captain) to P (the most junior pirate). Being an unusually democratic band of pirates, they use the following system. The captain proposes a division of the coins i.e., the number of coins each pirate will receive, and all the pirates (including the caption) vote yes or no on the captain's proposal. If there are at least as many yes votes as no votes, the proposal is accepted. Otherwise, the captain is made to walk the plank, the next most senior pirate on the ship becomes captain, and the process starts again from the beginning.

The pirates are also highly skilled logicians, and will make proposals and vote in a way that will give them the most preferred outcome. Firstly, let the *happiness* of a pirate be the number of coins he/she receives, or -1 if the pirate is made to walk the plank. An *outcome* is the list of happiness values for all the pirates. Given two outcomes, a pirate will compare them as follows:

1. He/she prefers the outcome in which he/she is happier.
2. If there is a tie, he/she prefers the outcome in which the most pirates walk the plank.
3. If there is still a tie, he/she identifies the most senior pirate whose happiness differs between the outcomes, and prefers the outcome in which this pirate is happier.

Task

The captain has been drinking too much rum, and is struggling to determine what the final outcome will be. Help him by writing a program to determine the happiness of each crew member when the coins are distributed.

Example

Suppose $S = 5$. If there are only two pirates, the captain will vote for herself, ensuring that her vote passes. Thus, she can safely vote all the coins for herself. With three pirates, the middle one will vote against any proposal by the captain, because if the captain walks the plank the middle one will be able to keep all the coins for herself. She will vote no even if the captain proposes to give her all the coins, due to the first tie-breaker rule.

However, if the captain proposes to keep four coins and give one to the most junior pirate, the vote will pass. The captain will obviously vote for it, and the junior pirate will vote for it since the alternative is to get no coins.

Input (stdin)

The input contains one line with two space-separated integers, S and P.

Sample input

```
5 3
```

Output (stdout)

The output contains P lines. The ith line contains the happiness of the ith pirate (from most senior to most junior) with the final outcome.

Sample output

```
4
0
1
```

Constraints

In all test cases, $1 \leq S$ and $1 \leq P$.

Subtask 1 (25 points)

- $S \leq 10$, $P \leq 10$

Subtask 2 (25 points)

- $S \leq 1\,000$, $P \leq 500$
- $P \leq S$

Subtask 3 (25 points)

- $S \leq 10\,000$, $P \leq 100$

Subtask 4 (25 points)

- $S \leq 10\,000$, $P \leq 3\,000$

South African Programming Olympiad
Day 1

Scoring

For each subtask you will receive either 100% or 0%. Your score will be the sum of your scores on the subtasks.

Time limit

2 seconds. Python: 20 seconds.

Detailed feedback

Detailed feedback is enabled for this problem. You are limited to 20 submissions with detailed feedback.

Word Math

Introduction

All pirates have treasure, and all treasure is hidden. All hidden treasure is marked on a map. However, not all treasure is marked with an X. To start his pirating career, Fred has acquired a map from a completely-honest-not-at-all-manic barkeep, Derf. Derf claims that the map shows the way to a vast sum of treasure. However, Fred does not see where the treasure is hidden on the map.

According to the bar tender, the key lies in the words scribed around the edges of the map. The words come in threes, of the form WORD + ANOTHER WORD = FINAL WORD. According to Derf, each letter of the words corresponds to a digit from zero to nine, and when the digits are substituted for the numbers, the sum is correct. No two letters in the sum correspond to the same digit. Leading zeroes in the solution numbers are not allowed. For instance, 008 is illegal.

Fred thinks that he may have been swindled as he thinks that some of the sums have multiple solutions, or even have no solution. Given the three words, help Fred work out how many different solutions there are to the sum.

Task

Write a program which, given three words, A, B and C, finds all numeric solutions to the problem: $A + B = C$. Output the *number* of numeric solutions matching the three words. If no solutions exist output 0.

Example

If, for example, Fred sees "A + A = BC", there are a total of five answers which are valid. These are:

$5 + 5 = 10$, $6 + 6 = 12$, $7 + 7 = 14$
$8 + 8 = 16$, $8 + 8 = 18$

Input (stdin)

The input will consist of three lines of capital English letters. These correspond to A, B and C respectively.

Sample input

```
A
A
BC
```

Output (stdout)

The output consists of one line consisting of the number of solutions.

Sample output

```
5
```

Constraints

If D is the number of distinct letters in A, B and C, and L is the maximum length of A, B and C, then

Subtask 1 (20 points)

- $D \leq 3$, $L \leq 4$

Subtask 2 (10 points)

- $D \leq 3$, $L \leq 7$

Subtask 3 (70 points)

- $D \leq 7$, $L \leq 8$

Scoring

Each subtask will consist of multiple test cases, each of which receives 100% for a correct solution and 0% for an incorrect solution.

Ambiguous Code

Introduction

Wyoming James, the moderately intrepid treasure hunter, has found directions to a pirate's treasure chest. The problem is the directions are encoded. Wyoming's managed to make some progress: he has narrowed down what each word can be to at most two possibilities, and knows that some pairs of letters match. Of course, it is possible he could have made a mistake and there could be no possible solution. Help Wyoming decode the directions to find the treasure.

Task

Given the list of possible words for each codeword and pairs of letters which match, determine whether the directions can be decoded, and if they can, decode them. There may be more than one decoding, but you need find only one.

A decoding consists of one of the possible words for each codeword. Letters which are listed as matching have to be the same.

Example

The message consists of three words, the first of which is either "TREASURE" or "TORTILLA", the second of which is either "NORTH" or "SOUTH" and the third of which is definitely "ISLAND", and Wyoming knows that the first letter of the first word is the same as the fourth letter of the second word, that the first letter of the second word is the same as the second letter of the third word, that the fourth letter of the first word is the same as the fourth letter of the third word and that the third letter of first word is the same as the eighth letter of the first word. The only possible message is "TREASURE SOUTH ISLAND".

Input (stdin)

The first line of input consists of two integers, M and N: the number of words and the number of matching pairs of letters respectively.

The next M lines each contain an integer K, which will always be 1 or 2, followed by K words. If $K = 2$, both words have the same length. The ith of these lines indicates the ith codeword can be decoded as any of the words on that line.

The final N lines list the pairs of matching letters. Each line consists of four integers — C_1, L_1, C_2, L_2 — indicating that the L_1th letter of word C_1 is the same as the L_2th letter of word C_2.

Sample input

```
3 4
2 TREASURE TORTILLA
2 NORTH SOUTH
1 ISLAND
1 1 2 4
2 1 3 2
1 4 3 4
1 3 1 8
```

Output (stdout)

The first line of output is `YES` if the directions can be decoded and `NO` otherwise.

If the directions can be decoded, the next M lines should each consist of a single word: the word which is the decoding of the corresponding codeword. Answers should be given in the same order as the words were given in the input.

Sample output

```
YES
TREASURE
SOUTH
ISLAND
```

Constraints

- $1 \leq$ length of each answer ≤ 20
- Each word consists of only uppercase A to Z.
- $1 \leq M, N$

Subtask 1 (20 points)

- $M \leq 16$, $N \leq 200$

Subtask 2 (50 points)

- $M, N \leq 1\,000$

Subtask 3 (30 points)

- $M, N \leq 100\,000$

Time limit

2 seconds. Python: 20 seconds.

Scoring

A correct `NO` will receive 100%. A correct `YES` will receive 30% and the remaining marks will be given for a fully correct solution.