The 2016 ACM ICPC Vietnam National Programming Contest

The problem set consists of **11** problems:

Problem **A**	ICPC score distribution
Problem **B**	Magical palindromic number
Problem **C**	Two matrices
Problem **D**	String again
Problem **E**	Counter terrorist
Problem **F**	K^{th} number
Problem **G**	Expected sum difference
Problem **H**	Chocolate
Problem **I**	Go home
Problem **J**	Spiral board
Problem **K**	Twin towers

Notes: contestants should use standard input and output for all problems.

Problem A: ICPC score distribution

After the provincial contests, the Vietnamese ACM ICPC judges and problem setters wants to analyze how good the problem set is. One criteria of a good problem set is the score distribution should be close to the normal distribution which graphically gives us a shape of a bell curve.

Given the problems solved by each team, your task is to visualize this data.

Input

The input starts with T - the number of tests, then T tests follow. Each test has the format:

- The first line is N - the number of teams ($N \leq 200$) and P - the number of problems ($P \leq 26$).
- The i-th line in the next N lines contains a string representing all problems solved by team i-th. Problems are indexed by first P uppercase letters.
- A team who did not solve any problem will be represented by an empty line.

Output

For each test, you need to print $P + 1$ lines followed by an empty line. The first $P + 1$ lines which visualize the distribution, has the format:

- X <space> S_X where S_X is a string contains only the asterisk character (*). The length of string S_X should be equal to the number of teams that solved X problems.
- X should be printed in increasing order.
- The number X should be aligned to the right (refer to sample output for clarification) and rightmost spaces should be trimmed.

Sample input	Sample output	Space clarification
2	0	0
13 7	1 *	1_*
ABCDEF	2 **	2_**
ABCDG	3 ***	3_***
ABCEF	4 ****	4_****
ABCD	5 **	5_**
ABCD	6 *	6_*
ABCE	7	7
ABCF		<empty line>
ABC	0 *	_0_*
ABC	1 *	_1_*
ACD	2 *	_2_*
AC	3	_3
AC	4 **	_4_**
A	5	_5
6 10	6	_6
ABCD	7	_7
EFGH	8 *	_8_*
	9	_9
ABCDEFGH	10	10
AB		<empty line>
J		

Note: Submissions to this problem are validated with `space_sensitive` flag. You need to output the exact number of spaces in order to get accepted. The third column shows the expected output with spaces replaced with character "_".

Problem B: Magical palindromic number

A number X is considered palindromic number, if it has the same value when written backward or forward. It is allowed to add leading zeros to the number to make it palindromic. For example:

- 7, 22, 12321 are palindromic numbers.
- 5050, 100 are palindromic numbers (can be re-written as 05050 and 00100).
- 103, 234 are not palindromic numbers.

A number X with both $X - 1$ and $X + 1$ being palindromic numbers is considered a magical palindromic number. Given 2 numbers L and R, your task is to count the number of magical palindromic numbers in the range $[L, R]$.

Input

The first line contains the only integer T - number of tests, in the next T lines, each line contains 2 numbers L and R $(1 \leq L \leq R \leq 10^{18})$.

Output

For each test case, print a single number - the number of magical palindromic numbers in the range $[L, R]$.

Sample input	Sample output
3 18 23 21 21 50 55	1 1 0

Problem C: Two matrices

You are given 2 matrices A and B, each with M rows and N columns. You are also given an integer number K. A sub-matrix of a matrix P consists of some consecutive rows and some consecutive columns of P.

Let S be a sub-matrix of matrix A, with m rows and n columns. We number the rows of the sub-matrix S from 1 to m, and the columns of the sub-matrix S from 1 to n. Let S_{ij} be the element at the i-th row and j-th column of sub-matrix S.

Let T be a sub-matrix of matrix B, also with m rows and n columns. Again, we number the rows of the sub-matrix T from 1 to m, and the columns of the sub-matrix T from 1 to n. Let T_{ij} be the element at the i-th row and j-th column of sub-matrix T.

Sub-matrix S is considered similar to sub-matrix T if $|S_{ij} - T_{ij}| \leq K$ for all pairs of i, j satisfy $1 \leq i \leq m, 1 \leq j \leq n$. Your task is to find 2 similar sub-matrices S and T with the maximal number of elements.

Input

The first line of the input contains the only integer T - the number of test cases. For each test case:

- The first line contains 3 integer numbers M, N and K. $(1 \leq M, N \leq 64, 0 \leq K \leq 10^9)$.
- The next M lines, each contains N space-separated numbers, representing matrix A.
- The next M lines, each contains N space-separated numbers, representing matrix B.
- All numbers in matrix A and B does not exceed 10^9 by absolute value.

Output

For each test case, print exactly one line, containing the maximum number of elements of matrix S (note that S and T has same number of elements).

Sample input	Sample output
2 3 3 0 1 1 1 1 1 1 1 1 2 2 2 2 2 1 1 2 1 1 3 3 1 1 1 1 1 1 1 1 1 1 2 2 2 2 1 1 2 1 1	4 9

Problem D: String again

You are given a list of N strings. A string is considered good if it is substring of at least K strings in your list. For example, $K = 2, N = 3$, and we have 3 strings: ABC, BBBBC, CCAACC:

- BC is a good string because it is substring of 2 strings: the 1st string and the 2nd string.
- AC is not a good string because it is substring of only the 3rd string.
- C is a good string because it is substring of all the 3 given strings.

Your task is to count the number of unique non-empty good strings.

Input

The input starts with the number of tests - T. Then T tests follow:

- The first line contains 2 numbers N and K $(2 \leq K \leq N \leq 50)$.
- The next N lines, each line is a string which only contains upper case letter and its length is at most $100{,}000$. The total length of N strings is at most $500{,}000$.

Output

For each test in the input, print the number of unique non-empty good strings.

Sample input	Sample output
1 3 2 ABC BBBBC CCACC	4

Explanation: The good strings are A, B, C and BC.

Problem E: Counter terrorist

The intelligence department has just captured an encrypted message between 2 known terrorists and they want to decrypt this message. There are N known decryption methods, method i will take T_i seconds to execute. But in order to successfully decrypt a message, they might need to combine multiple different methods. Combining methods $m_1, m_2, \dots m_K$ will take $T_{m_1} * T_{m_2} * \dots * T_{m_K}$ seconds to execute. Their strategy to the decryption process is:

- Step 1: Trying all individual method
- Step 2: Trying to combine 2 methods
- Step k: Trying to combine k methods
- Step N: Trying to combine all N methods

Before actually executing this process, they want to examine how much time it will take to execute each step.

Input

The input starts with the number of tests - T. Then T tests follow. Each test has format:

- The first line contains the number N. $(N \leq 50000)$
- The second line consists of N positive integers $T_1, T_2, \dots, T_N$. $(T_i \leq 10^9)$

Output

For each test in the input, print in a single line containing N integers separated by a single space. The i-th integer denotes the time needed to execute step i modulo $10{,}007$.

Sample input	Sample output
2 3 1 2 3 2 19 9	6 11 6 28 171

Problem F: K^{th} number

Given 3 arrays of integers A, B, C, with length n_A, n_B, n_C, respectively. The arrays are 1-based indexed. We generate a new array X using the following pseudo code:

```
nX = 0
X = []   // initialize X to empty array
for i = 1 to nA
     for j = 1 to nB
          for k = 1 to nC
               nX += 1
               X[nX] = A[i] * B[j] * C[k]
          endfor
     endfor
endfor
```

Given an integer number K, your task is to calculate the K-th smallest number in array X.

Input

The input starts with the number of test – T, then T tests follow. Each test has format:

- The first line contains 4 integers n_A, n_B, n_C and K. $(1 \leq n_A, n_B, n_C \leq 1000;\ K \leq \min(10^6, n_A \times n_B \times n_C))$.
- The next line contains n_A integer numbers representing array A.
- The next line contains n_B integer numbers representing array B.
- The next line contains n_C integer numbers representing array C.
- All the numbers in array A, B, C does not exceed 1000 in absolute value.

Output

For each test case, print one single number - the K-th smallest number in array X.

Sample input	Sample output
1 2 3 2 2 3 2 -2 1 -1 0 2	-8

Problem G: Expected sum difference

RR has a board with 2 rows and N columns. RR will use $2N$ numbers from 1 to $2N$ to fill in this board. The locations of these numbers on the board are random. More precisely, for each number from 1 to $2N$, RR will put it in one random, empty position on the board.

Let array B be the first row of the board and array C be the second row of the board. Note that both arrays will have exactly N numbers. Let $F = \sum_{i=1 \ldots N} |B_i - C_i|$. Your task is to calculate the expected value of F.

Input

The input starts with the number of test - T ($T \leq 1000$). Then T tests follow. Each test contains a single integer N. $(1 \leq N \leq 10^9)$

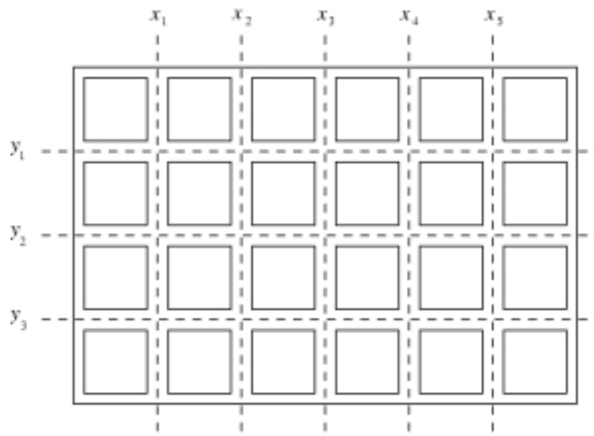
Output

For each test in the input, print the answer with **exactly** 2 digits after decimal place.

Sample input	Sample output
2 1 2	1.00 3.33

Problem H: Chocolate

Brothers Bo and Bi are given a chocolate bar for Halloween. The chocolate bar composed of $m \times n$ square pieces. One may break the chocolate bar along the vertical and horizontal lines as indicated by the broken lines in the picture. Breaking along the vertical line x_1 will result in 2 portions: 4 pieces and 20 pieces. If we break along the vertical line x_3 or the horizontal line y_2, both ways will result in 2 portions: 12 pieces and 12 pieces.



The parents want Bo and Bi to share this chocolate bar. But Bo is a greedy older brother, he breaks once and always takes the larger portion (if 2 portions are different). Given the size m, n of the chocolate bar, your task is to list all the possibilities of Bo's portion in increasing order.

Input

The input starts with the number of tests - T, then T tests follow. Each test is printed in one line consists of 2 positive integers m and n $(2 < m + n \leq 20)$.

Output

For each test, print all the possibilities of Bo's portion in increasing order.

Sample input	Sample output
2 1 3 4 6	2 12 16 18 20

Problem I: Go home

Row, row, row a boat, gently down the stream.

Merrily, merrily, merrily, merrily, life is but a dream.

Every day, Hieu must go to school. Hieu's house is at point A with coordinates (x_A, y_A). Hieu's school is at point B with coordinates (x_B, y_B). There is a river flowing in a straight line, from $(-10^9, 0)$ to $(10^9, 0)$. In the river, water flows with speed v_S. Hieu can walk in land with speed v_W or row a boat in the river with speed v_R (relative to the water). Your task is to calculate what is the minimum time for Hieu to go from home to school and from school to home?

Input

The input starts with the number of test – T $(T \le 100)$. Then T tests follow. Each test is printed in a single line contains 7 space-separated integers: $x_A, y_A, x_B, y_B, v_S, v_W, v_R$.

All numbers in input does not exceed 100 by absolute value and $v_S, v_W, v_R > 0$.

Output

For each test, print one number: the total time for Hieu to go from home to school and from school to home, with **exactly** 2 digits after decimal place.

Sample input	Sample output
1 0 2 8 -4 1 10 8	2.00

Problem J: Spiral board

Tung has a board with N rows and N columns. The rows are numbered 1 to N from top to bottom, the columns are numbered 1 to N from left to right and cell (x, y) is correspondent to the cell in x-th row, y-th column. Tung fills this board with numbers from 1 to $N \times N$ in spiral form starting at cell $(1\ 1)$. He goes inward in clockwise direction. The table below shows how he fills a 5×5 board. Using this board, Tung wants to transform a number X to a number Y by performing several moves. In each move, he can change the current number X to a new number Z if X and Z are coprime and the cell containing the number X and the cell containing the number Z share an edge. Given N, X, Y; Tung wants to know what is the minimal number of moves he needs to perform in order to transform X into Y.

1	2	3	4	5
16	17	18	19	6
15	24	25	20	7
14	23	22	21	8
13	12	11	10	9

Input

The input starts with the number of test - T. Then T tests follow. Each test consists of 3 integers N, X, Y. $(1 \leq N \leq 1000; 1 \leq X, Y \leq N^2)$

Output

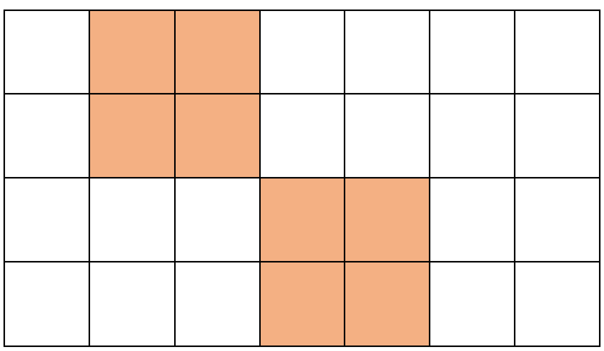
For each test, print the minimal number of transformations Tung needs to take. If it is impossible to transform, print -1 instead.

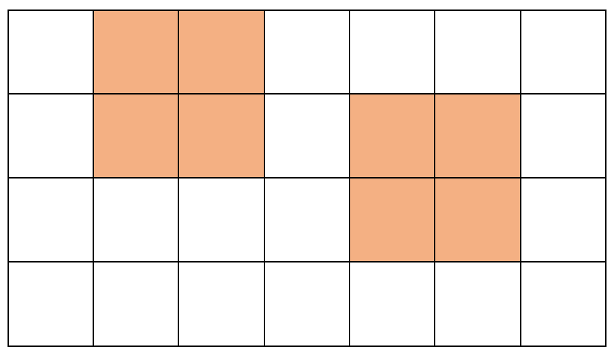
Sample input	Sample output
1 5 3 18	3

Problem K: Twin towers

Ho Chi Minh city is planning its new urban redevelopment project in district 2. The new urban area will cover an area of 7 square kilometers and consists of several sections: office building, parks and residence. This area is expected to become a great new financial district of the city.

The highlight of this area will be the twin towers. The city approved the plan for building these iconic towers in a rectangular field which can be divided into $M \times N$ cells. The M rows are numbered from 1 to M, the N rows are numbered from 1 to N. Each tower will cover an area of $K \times K$ cells and must not touch each other. In the first example, the twin towers cannot cover these 2 orange areas because they share a single point. While in the second example, these 2 areas might be used to build the twin towers.





In the field, there are several cells which are better than the others; in terms of technical reasons like stronger platform, water resources, etc. They want to take advantage of all these good cells and build the towers on top of these.

Given the size of the field, the size of each tower and the location of the good cells, your task is to calculate how many different locations to build the twin towers.

Input

The input starts with the number of test – T. Then T tests follow. Each test has the format:

- The first line consists of 4 integers M, N, K and W where W is the number of good cells. $(2 \le M, N \le 200;\ 1 \le K \le min(M, N);\ 0 \le W \le 20)$
- The next W lines contains a pair of integers (r, c) describing the location of the good cells. $(1 \le r \le M, 1 \le c \le N)$

Output

For each test, print the number of possibilities to build the twin towers. Please note that, if the top-left cells of the 2 towers locate at (a, b) and (c, d), this solution will be considered similar to (c, d) and (a, b).

Sample input	Sample output
2 4 7 2 3 2 2 2 3 2 5 6 6 2 2 2 2 5 5	4 15