The 39th ACM International Collegiate Programming Contest

Asia Mudanjiang Regional Contest

Contest Section

Problem List

A	Average Score
B	Building Fire Stations
C	Card Game
D	Domination
E	Excavator Contest
F	Fiber-optic Network
G	Garden and Sprinklers
H	Hierarchical Notation
I	Information Entropy
J	Jacobi Pattern
K	Known Notation

This problem set should contain 11 (eleven) problems on 24 (twenty-four) numbered pages. Please inform a runner immediately if something is missing from your problem set.

Problem A. Average Score

Description

Bob is a freshman in Marjar University. He is clever and diligent. However, he is not good at math, especially in Mathematical Analysis.

After a mid-term exam, Bob was anxious about his grade. He went to the professor asking about the result of the exam. The professor said:

"Too bad! You made me so disappointed."

"Hummm... I am giving lessons to two classes. If you were in the other class, the average scores of both classes will increase."

Now, you are given the scores of all students in the two classes, except for the Bob's. Please calculate the possible range of Bob's score. All scores shall be integers within [0, 100].

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains two integers N ($2 <= N <= 50$) and M ($1 <= M <= 50$) indicating the number of students in Bob's class and the number of students in another class respectively.

The next line contains N - 1 integers A_1, A_2, .., A_{N-1} representing the scores of other students in Bob's class.

The last line contains M integers B_1, B_2, .., B_M representing the scores of students in another class.

Output

For each test case, output two integers representing the minimal possible score and the maximal possible score of Bob.

It is guaranteed that the solution always exists.

Sample Input	Output for the Sample Input
2 4 3 5 5 5 4 4 3 6 5	4 4 2 4

Sample Input	Output for the Sample Input
5 5 4 5 3 1 3 2 2 1	

Problem B. Building Fire Stations

Description

Marjar University is a beautiful and peaceful place. There are N buildings and N - 1 bidirectional roads in the campus. These buildings are connected by roads in such a way that there is exactly one path between any two buildings. By coincidence, the length of each road is 1 unit.

To ensure the campus security, Edward, the headmaster of Marjar University, plans to setup two fire stations in two different buildings so that firefighters are able to arrive at the scene of the fire as soon as possible whenever fires occur. That means the longest distance between a building and its nearest fire station should be as short as possible.

As a clever and diligent student in Marjar University, you are asked to write a program to complete the plan. Please find out two proper buildings to setup the fire stations.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains an integer N ($2 <= N <= 200000$).

For the next N - 1 lines, each line contains two integers X_i and Y_i. That means there is a road connecting building X_i and building Y_i (indexes are 1-based).

Output

For each test case, output three integers. The first one is the minimal longest distance between a building and its nearest fire station. The next two integers are the indexes of the two buildings selected to build the fire stations.

If there are multiple solutions, any one will be acceptable.

Sample Input	Output for the Sample Input
2 4 1 2 1 3 1 4 5	1 1 2 1 2 4

Sample Input	Output for the Sample Input
1 2 2 3 3 4 4 5	

Problem C. Card Game

Description

Earthstone is a famous online card game created by *Lizard Entertainment*. It is a collectible card game that revolves around turn-based matches between two opponents. Players start the game with a substantial collection of basic cards, but can gain rarer and more powerful cards through purchasing packs of additional cards, or as rewards for competing in the arena. Card packs can be purchased with gold, an in-game currency rewarded for completing random daily quests and winning matches, or by using real money in the in-game store.

Edward is the headmaster of Marjar University. He has lots of money and lots of spare time. Recent days, he was addicted to *Earthstone* that he bought over 400 card packs! After having purchased so many cards, he believed that he was invincible. Then, he entered the arena and started a match... Before we continue the story of Edward, let us look at the rules of *Earthstone* first.

Each *Earthstone* battle is a one on one turn-based match between two opponents. During a player's turn, he can choose to play any of his cards and command the minions to attack targets. Those played cards will be placed on the table as they are 'summoned' as minions. Each card has two basic attributes:

- Attack A_i: If a minion attacks a character or was attacked, it will deal A_i points of damage to the opponent. A character whose attack value is zero cannot actively attack.
- Health H_i: The minion has H_i points of initial health. After being damaged, the minion's health will decrease by the corresponding damage value. The minion will be killed and discarded if its health is less than or equal to zero.

If a minion attacks another minion, both of them will receive damage simultaneously. Besides minions, each player has a hero with some points of initial health. The hero has zero points of attack value and does not have any skills. If a player's hero is killed, he will lose the game. By the way, the word 'character' in *Earthstone* can mean either a hero or a minion.

Apart from the two basic attributes, minions may have zero or more following abilities:

- Charge: A minion cannot actively attack on the same turn that it is summoned, unless it has the Charge ability.
- DivineShield: Absorbs the first time of non-zero damage taken by the minion, removing the shield.
- Taunt: Enemies must attack minions with Taunt before any non-Taunt characters.
- Windfury: During a player's turn, the player can command all his existing minions or newly summoned Charge minions to attack enemy characters for at most one time. But if a minion has the Windfury ability, it can attack twice instead of once.

Now, let us come back to the story of Edward. Edward is playing the game and it is Edward's turn now. There are already $X + Y$ minions on the table. X of them are on the Edward's side and the other minions are fighting for the opponent. Besides the existing minions, Edward has Z cards in his hand. The health of opponent's hero is M. There is no limitation about the number of cards can be played on a turn or the maximal number of minions on the table.

Edward is a cautious man, he wants you find out a long-term game strategy about the current situation. You need to reduce the potential damage received on the next turn (the opponent's turn). The potential damage is defined as the sum of attack value of the opponent's minions, with Windfury minions count twice. If there are multiple solutions, please find out the solution which can deal as many as possible points of damage to the opponent's hero. But, Edward will also not miss any chance to win the game. If there exists a strategy to kill the opponent's hero within the current turn, he will use the winning strategy instead.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains 4 integers X, Y, Z ($0 <= X + Z <= 8$, $0 <= Y <= 15$) and M ($1 <= M <= 100$).

The following lines of input consists of 3 parts. These parts have the same input format and contain X, Y, Z lines respectively. Each line describes a minion or a card, in the format of "A_i/H_i *abilities*" ($0 <= A_i <= 12$, $1 <= H_i <= 15$). The *abilities* contains zero or more ability names, separated by a space. There is no space after H_i if the *abilities* is empty.

Output

For each test case, output the minimal potential damage and the maximal damage that Edward can deal to the opponent's hero.

If the opponent's hero can be killed within the current turn, output "Well played" instead (without quotes).

Sample Input

```
2
2 3 1 30
2/1 Charge
9/5
6/5 Taunt
4/2 DivineShield Charge
6/7
3/5 Windfury Charge DivineShield Taunt
0 4 1 1
5/2 Charge
3/3 DivineShield
```

Output for the Sample Input

```
4 3
Well played
```

Sample Input	Output for the Sample Input
6/7 4/5 Windfury 1/1 Charge	

Hint

There are large amount of small test cases and a few number of big test cases.

Problem D. Domination

Description

Edward is the headmaster of Marjar University. He is enthusiastic about chess and often plays chess with his friends. What's more, he bought a large decorative chessboard with N rows and M columns.

Every day after work, Edward will place a chess piece on a random empty cell. A few days later, he found the chessboard was *dominated* by the chess pieces. That means there is at least one chess piece in every row. Also, there is at least one chess piece in every column.

"That's interesting!" Edward said. He wants to know the expectation number of days to make an empty chessboard of $N \times M$ dominated. Please write a program to help him.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

There are only two integers N and M ($1 <= N, M <= 50$).

Output

For each test case, output the expectation number of days.

Any solution with a relative or absolute error of at most 10^{-8} will be accepted.

Sample Input	Output for the Sample Input
2 1 3 2 2	3.000000000000 2.666666666667

Problem E. Excavator Contest

Description

Bluefly University is famous of excavator technology. Lots of students take part in many excavator-related courses. After the students finish their courses, they will compete in a contest called International Collegiate Excavator Contest (ICEC).

This year's ICEC will be held at Marjar University. This is an individual competition that each contestant will start the match one by one.

The task of the contest is to drive an excavator passing a square field. The judge partitioned the field into $N \times N$ equal-sized square chunks. Each chunk should be visited exactly one time. The contestant will drive the excavator, starting from and ending at the center of two different boundary chunks.

In order to show off their superb excavator operating skills, the contestants need to drive the excavator with as many as possible turnings. Since the excavator is a kind of large and heavy vehicle, it can only make a turn to left or right at the center of any chunk.

Bob is a student from Marjar University. He wants to win the contest. To fulfill this dream, he needs to drive the excavator with at least $N \times (N - 1) - 1$ turnings. It seems to be a difficult task, so he turns to you for help. Please write a program to find a feasible route for him.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

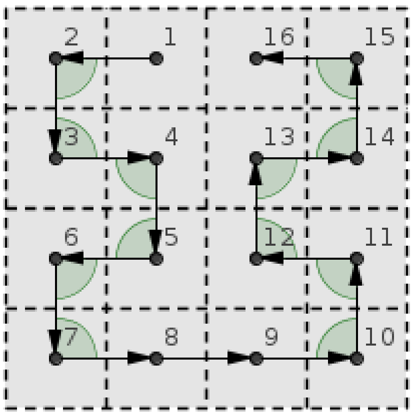
There is only one integer N ($2 <= N <= 512$).

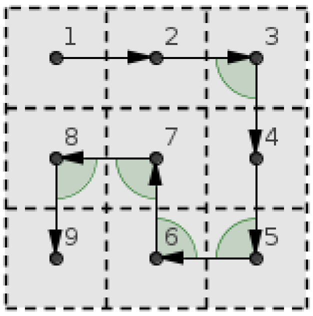
Output

For each test case, output a matrix with $N \times N$ integers indicating the route to pass all the chunks. Bob will drive the excavator passing the chunks in the order of 1, 2, .., N^2. If there are multiple solutions, any one will be acceptable.

Sample Input	Output for the Sample Input
2 4 3	2 1 16 15 3 4 13 14 6 5 12 11 7 8 9 10 1 2 3 8 7 4 9 6 5

Hint





Problem F. Fiber-optic Network

Description

Marjar University has decided to upgrade the infrastructure of school intranet by using fiber-optic technology. There are N buildings in the school. Each building will be installed with one router. These routers are connected by optical cables in such a way that there is exactly one path between any two routers.

Each router should be initialized with an operating frequency F_i before it starts to work. Due to the limitations of hardware and environment, the operating frequency should be an integer number within $[L_i, R_i]$. In order to reduce the signal noise, the operating frequency of any two adjacent routers should be co-prime.

Edward is the headmaster of Marjar University. He is very interested in the number of different ways to initialize the operating frequency. Please write a program to help him! To make the report simple and neat, you only need to calculate the sum of F_i (modulo 1000000007) in all solutions for each router.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains one integer N ($1 <= N <= 50$). The next line contains N integers L_i ($1 <= L_i <= 50000$). Then, the following line contains N integers R_i ($L_i <= R_i <= 50000$).

For the next N - 1 lines, each line contains two integers X_i and Y_i. That means there is an optical cable connecting router X_i and router Y_i (indexes are 1-based).

Output

For each test case, output a line with N integers representing the sum of F_i (modulo 1000000007) in all solutions.

Sample Input	Output for the Sample Input
2 4 1 2 3 4 2 3 4 5 1 2 2 3	5 10 14 19 10 23 31 41

Sample Input	Output for the Sample Input
3 4 4 1 2 3 4 2 3 4 5 1 2 1 3 1 4	

Hint

In the first sample test case, there are 4 ways to initialize the operating frequency:

- 1 2 3 4
- 1 2 3 5
- 1 3 4 5
- 2 3 4 5

Problem G. Garden and Sprinklers

Description

There is a beautiful garden in Marjar University. Recently, Edward, the headmaster of Marjar University, decided to build a garden water sprinkler system. The system consists of three sprinklers.

Assuming that Marjar University is an infinite plane, the garden is a circle whose center is at (X_0, Y_0) with radius R. Now, Edward has already determined the position for two sprinklers at (X_1, Y_1) and (X_2, Y_2). He needs to choose the position for the last sprinkler. Here are some conditions to be satisfied:

1. The three sprinklers should not in the same line.
2. The last sprinkler should be located inside or on the boundary of the garden.
3. The coordinates of the sprinklers must be integers.
4. Twice the area of the triangle that the three sprinklers form should equals S.

Under these conditions, Edward wants to know the number of possible positions for the last sprinkler. Please write a program to help him!

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains an integer S ($1 <= S <= 10^8$). The next line contains three integers X_0, Y_0 and R ($1 <= R <= 10^8$). The last line contains four integers X_1, Y_1, X_2 and Y_2.

It is guaranteed that the absolute value of all input coordinates will not exceed 10^8 and the positions of the two existing sprinklers are different.

Output

For each test case, output the number of possible positions.

Sample Input	Output for the Sample Input
1 4 0 0 4 -1 0 1 0	14

Hint

In the sample test case, the possible positions for the last sprinkler are: (-3, 2), (-2, 2), (-1, 2), (0, 2), (1, 2), (2, 2), (3, 2), (-3, -2), (-2, -2), (-1, -2), (0, -2), (1, -2), (2, -2), (3, -2).

Problem H. Hierarchical Notation

Description

In Marjar University, students in College of Computer Science will learn EON (Edward Object Notation), which is a hierarchical data format that uses human-readable text to transmit data objects consisting of attribute-value pairs. The EON was invented by Edward, the headmaster of Marjar University.

The EON format is a list of key-value pairs separated by comma ",", enclosed by a couple of braces "{" and "}". Each key-value pair has the form of "<key>":"<value>". <key> is a string consists of alphabets and digits. <value> can be either a string with the same format of <key>, or a nested EON.

To retrieve the data from an EON text, we can search it by using a key. Of course, the key can be in a nested form because the value may be still an EON. In this case, we will use dot "." to separate different hierarchies of the key.

For example, here is an EON text:

{"headmaster":"Edward","students":{"student01":"Alice","student02":"Bob"}}

- For the key "headmaster", the value is "Edward".
- For the key "students", the value is {"student01":"Alice","student02":"Bob"}.
- For the key "students"."student01", the value is "Alice".

As a student in Marjar University, you are doing your homework now. Please write a program to parse a line of EON and respond to several queries on the EON.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains an EON text. The number of colons ":" in the string will not exceed 10000 and the length of each key and non-EON value will not exceed 20.

The next line contains an integer Q ($0 <= Q <= 1000$) indicating the number of queries. Then followed by Q lines, each line is a key for query. The querying keys are in correct format, but some of them may not exist in the EON text.

The length of each hierarchy of the querying keys will not exceed 20, while the total length of each querying key is not specified. It is guaranteed that the total size of input data will not exceed 10 MB.

Output

For each test case, output Q lines of values corresponding to the queries. If a key does not exist in the EON text, output "Error!" instead (without quotes).

Sample Input

```
1
{"hm":"Edward","stu":{"stu01":"Alice","stu02":"Bob"}}
4
"hm"
"stu"
"stu"."stu01"
"students"
```

Output for the Sample Input

```
"Edward"
{"stu01":"Alice","stu02":"Bob"}
"Alice"
Error!
```

Problem I. Information Entropy

Description

Information Theory is one of the most popular courses in Marjar University. In this course, there is an important chapter about information entropy.

Entropy is the average amount of information contained in each message received. Here, a message stands for an event, or a sample or a character drawn from a distribution or a data stream. Entropy thus characterizes our uncertainty about our source of information. The source is also characterized by the probability distribution of the samples drawn from it. The idea here is that the less likely an event is, the more information it provides when it occurs.

Generally, "entropy" stands for "disorder" or uncertainty. The entropy we talk about here was introduced by Claude E. Shannon in his 1948 paper "A Mathematical Theory of Communication". We also call it Shannon entropy or information entropy to distinguish from other occurrences of the term, which appears in various parts of physics in different forms.

Named after Boltzmann's H-theorem, Shannon defined the entropy H (Greek letter H, η) of a discrete random variable X with possible values $\{x_1, x_2, ..., x_n\}$ and probability mass function $P(X)$ as:

$$\mathrm{H(X)} = \mathrm{E}(-\ln(\mathrm{P}(x_i)))$$

Here E is the expected value operator. When taken from a finite sample, the entropy can explicitly be written as

$$\mathrm{H(X)} = -\sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{P}(x_i) \log_{\mathrm{b}} \mathrm{P}(x_i)$$

Where b is the base of the logarithm used. Common values of b are 2, Euler's number e, and 10. The unit of entropy is *bit* for $b = 2$, *nat* for $b = e$, and *dit* (or digit) for $b = 10$ respectively.

In the case of $P(x_i) = 0$ for some i, the value of the corresponding summand $0 \log_b(0)$ is taken to be a well-known limit:

$$0 \log_{\mathrm{b}}(0) = \lim_{\mathrm{p} \to 0+} \mathrm{p} \log_{\mathrm{b}}(\mathrm{p})$$

Your task is to calculate the entropy of a finite sample with N values.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains an integer N ($1 <= N <= 100$) and a string S. The string S is one of "bit", "nat" or "dit", indicating the unit of entropy.

In the next line, there are N non-negative integers P_1, P_2, .., P_N. P_i means the probability of the i-th value in percentage and the sum of P_i will be 100.

Output

For each test case, output the entropy in the corresponding unit.

Any solution with a relative or absolute error of at most 10^{-8} will be accepted.

Sample Input

```
3
3 bit
25 25 50
7 nat
1 2 4 8 16 32 37
10 dit
10 10 10 10 10 10 10 10 10 10
```

Output for the Sample Input

```
1.500000000000
1.480810832465
1.000000000000
```

Problem J. Jacobi Pattern

Description

The College of Biological Sciences of Marjar University is famous for genetic engineering. After years of research, the scientists of Marjar University find some facts about genes.

1. Genes are composed of simpler units called nucleotides.
2. The kinds of nucleotides are limited. For now, there are only M kinds nucleotides.
3. For different genes, the arrangements of nucleotides are different. So the functions of the genes are also different.

Knowing the facts, the scientists use the arrangements of nucleotides to identify genes, called genome sequence. For example {1, 2, 3, 4} is a gene composed of 1-th, 2-th, 3-th and 4-th nucleotide.

However, Edward, a weird but genius scientist (he is also the headmaster of Marjar University), found that even for two different genes, their functions may be same. Edward was curious about it and studied many samples. Finally, he found the reason and called it the similarity of genes. Two genes are similar to each other, if and only if their genome sequences meet the following rules:

1. They have the same length of their genome sequence.
2. Their genome sequences are cyclically equivalent.

During the research, Edward found that if two similar genome sequences are concatenated, then it forms a pattern called Jacobi pattern (This pattern was in memory of the great work of the mathematician K. G. J. Jacobi). For example, if {1, 2, 3, 4} and {3, 4, 1, 2} are concatenated, we will have {1, 2, 3, 4, 3, 4, 1, 2}, which forms a Jacobi pattern. Now Edward gives you a genome sequence $\{A_1, A_2, ..., A_N\}$, he wants to know how many continuous subsequence of $\{A_1, A_2, .., A_N\}$ form a Jacobi pattern.

Please note, two sequences are cyclically equivalent if one of them can be obtained from another by moving its certain suffix from the end of the string to its beginning. For example {1, 2, 1, 2, 2, 1} and {1, 2, 2, 1, 1, 2} are cyclically equivalent, whereas {1, 2, 1, 2, 1} and {1, 1, 2, 2, 1} are not. In particular, every sequence is cyclically equivalent to itself.

Input

There are multiple test cases. The first line of input contains an integer T indicating the number of test cases. For each test case:

The first line contains two integers N and M ($1 <= N, M <= 5000$) indicating the length of the sequence and the number of kinds of nucleotides.

The following line contains N integers A_1, A_2, .., A_N ($1 <= A_i <= M$) indicating the given genome sequence.

Output

For each test case, output two integers X and Y on the first line, which indicates the total number of Jacobi patterns and the number of different length of Jacobi patterns.

For the next Y lines, firstly output two integers L_i and C_i, indicating a length of Jacobi patterns and the number of Jacobi patterns at this length. Then output C_i integers P_{i1}, P_{i2}, .., P_{iCi} in increasing order on the same line, where P_{ij} means there is a Jacobi pattern of length L_i starting from index P_{ij} (indexes are 1-based). These lines should be listed in the increasing order of L_i.

Sample Input	Output for the Sample Input
2 12 4 1 1 1 2 3 4 3 4 1 2 2 1 3 4 1 2 4	6 3 2 3 1 2 10 4 2 5 9 8 1 3 0 0

Hint

In the first sample test case, there are three Jacobi patterns of length 2: {1, 1}, {1, 1}, {2, 2}, two Jacobi patterns of length 4: {3, 4, 3, 4}, {1, 2, 2, 1} and only one Jacobi pattern of length 8: {1, 2, 3, 4, 3, 4, 1, 2}.

In the second sample test case, we cannot find any Jacobi pattern.

Problem K. Known Notation

Description

Do you know reverse Polish notation (RPN)? It is a known notation in the area of mathematics and computer science. It is also known as postfix notation since every operator in an expression follows all of its operands. Bob is a student in Marjar University. He is learning RPN recent days.

To clarify the syntax of RPN for those who haven't learnt it before, we will offer some examples here. For instance, to add 3 and 4, one would write "3 4 +" rather than "3 + 4". If there are multiple operations, the operator is given immediately after its second operand. The arithmetic expression written "3 - 4 + 5" in conventional notation would be written "3 4 - 5 +" in RPN: 4 is first subtracted from 3, and then 5 added to it. Another infix expression "5 + ((1 + 2) × 4) - 3" can be written down like this in RPN: "5 1 2 + 4 × + 3 -". An advantage of RPN is that it obviates the need for parentheses that are required by infix.

In this problem, we will use the asterisk "*" as the only operator and digits from "1" to "9" (without "0") as components of operands.

You are given an expression in reverse Polish notation. Unfortunately, all space characters are missing. That means the expression are concatenated into several long numeric sequence which are separated by asterisks. So you cannot distinguish the numbers from the given string.

You task is to check whether the given string can represent a valid RPN expression. If the given string cannot represent any valid RPN, please find out the minimal number of operations to make it valid. There are two types of operation to adjust the given string:

1. Insert. You can insert a non-zero digit or an asterisk anywhere. For example, if you insert a "1" at the beginning of "2*3*4", the string becomes "12*3*4".
2. Swap. You can swap any two characters in the string. For example, if you swap the last two characters of "12*3*4", the string becomes "12*34*".

The strings "2*3*4" and "12*3*4" cannot represent any valid RPN, but the string "12*34*" can represent a valid RPN which is "1 2 * 34 *".

Input

There are multiple test cases. The first line of input contains an integer *T* indicating the number of test cases. For each test case:

There is a non-empty string consists of asterisks and non-zero digits. The length of the string will not exceed 1000.

Output

For each test case, output the minimal number of operations to make the given string able to represent a valid RPN.

Sample Input	Output for the Sample Input
3 1*1 11*234** *	1 0 2