



ACM International Collegiate Programming Contest 2018

Latin American Regional Contests

November 9th-10th, 2018

Contest Session

This problem set contains 13 problems; pages are numbered from 1 to 16.

This problem set is used in simultaneous contests hosted in the following countries:

Argentina, Bolivia, Brasil, Chile, Colombia, Costa Rica, Cuba, El Salvador
México, Panamá, Perú, República Dominicana and Venezuela

General information

Unless otherwise stated, the following conditions hold for all problems.

Program name

1. Your solution must be called `codename.c`, `codename.cpp`, `codename.java`, `codename.py2` or `codename.py3`, where *codename* is the capital letter which identifies the problem.

Input

1. The input must be read from standard input.
2. The input consists of a single test case, which is described using a number of lines that depends on the problem. No extra data appear in the input.
3. When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input. There are no empty lines.
4. The English alphabet is used. There are no letters with tildes, accents, diaereses or other diacritical marks (ñ, Å, é, Ì, ô, Ü, ç, etcetera).
5. Every line, including the last one, has the usual end-of-line mark.

Output

1. The output must be written to standard output.
2. The result of the test case must appear in the output using a number of lines that depends on the problem. No extra data should appear in the output.
3. When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output. There should be no empty lines.
4. The English alphabet must be used. There should be no letters with tildes, accents, diaereses or other diacritical marks (ñ, Å, é, Ì, ô, Ü, ç, etcetera).
5. Every line, including the last one, must have the usual end-of-line mark.
6. To output real numbers, round them to the closest rational with the required number of digits after the decimal point. Test case is such that there are no ties when rounding as specified.

Problem A – A Symmetrical Pizza

Author: Mario Silva, Brasil

Bob has a symmetry craze. Everything in his life must be symmetric: his house, his clothes, his car, even his food. And pizza is no exception. For him to eat a pizza, all the toppings, like tomatoes, olives, pepperoni or basil, must be arranged with some degree of rotational symmetry.

This evening Bob ordered some pizza. As usual, when it arrived, he asked the delivery driver to demonstrate that the pizza met his demands for rotational symmetry. The driver demonstrated the symmetry then, as they are trained to do, using the following procedure:

- take a picture of the pizza with a cellphone;
- rotate the pizza by R degrees around its center;
- take another picture;
- show Bob the two pictures side by side, so that he sees that the pizza appears identical in both.

Satisfied, Bob paid for the pizza and took it to the kitchen. In order to test his brand new laser pizza cutter, he decided to cut the pizza in as many slices as possible. Of course, Bob wants to cut the slices in a way that all of them look exactly the same, in accordance with another of his crazes. Now given the angle R of the symmetry demonstration, Bob wants to know the maximum amount of equal slices he can cut the pizza in.

Input

The input consists of a single line that contains a rational number R ($0 < R < 360$) indicating the angle of the rotational symmetry demonstration. This number has exactly two digits after the decimal point.

Output

Output a single line with an integer representing the maximum amount of equal slices Bob can cut the pizza in, based on the provided information.

Sample input 1 45.00	Sample output 1 8
Sample input 2 180.00	Sample output 2 2
Sample input 3 240.00	Sample output 3 3
Sample input 4 35.00	Sample output 4 72
Sample input 5 2.50	Sample output 5 144
Sample input 6 11.34	Sample output 6 2000

Problem B – Building a Field

Author: Ricardo Anido, Brasil

John is a meticulous person. In his farm he built a circular field with some trees planted right at the circumference of the field. Figure (a) below shows the field with the trees.

Now John wants to use a long rope and four of the field trees to demarcate a rectangle using the trees as vertices and the rope as edges. Figure (b) below shows two rectangles that can be demarcated using the trees of the field in figure (a).



Given the description of the positions of the trees in John’s circular field, you must determine whether it is possible to demarcate a rectangle using four of the trees as vertices and the rope as edges.

Input

The first line contains an integer N ($4 \leq N \leq 10^5$) indicating the number of trees in the field. Trees are represented as points on a circumference. The second line contains N integers L_1, L_2, \dots, L_N ($1 \leq L_i \leq 10^6$ for $i = 1, 2, \dots, N$) indicating the arc lengths between each pair of consecutive trees. The arcs are given in counter-clockwise order. The total length of the circumference does not exceed 10^9 .

Output

Output a single line with the uppercase letter “Y” if it is possible to demarcate a rectangle using the given trees, and the uppercase letter “N” otherwise.

<p>Sample input 1</p> <p>8 3 3 4 2 6 2 2 2</p>	<p>Sample output 1</p> <p>Y</p>
<p>Sample input 2</p> <p>4 14 16 15 15</p>	<p>Sample output 2</p> <p>N</p>
<p>Sample input 3</p> <p>6 3 7 7 3 10 10</p>	<p>Sample output 3</p> <p>Y</p>

Problem C – Cheap Trips

Author: Alejandro Strejilevich de Loma, Argentina

Nlogonia has a new scheme for public transportation. When the first trip of a passenger starts, it also starts a 120 minutes interval such that discounts are applied to some of the trips that the passenger starts *before* the end of the interval. The discount for the second trip is 50% of the regular cost, while the discount for each of the remaining trips up to the sixth trip (that is, four more trips) is 75% of the regular cost. Once the 120 minutes interval ends, a new trip starts a new interval having the same kind of discounts.

Ástor is an exchange student that has just arrived to Nlogonia. He wants to spend as little money as possible for making a sequence of trips. The first trip in the sequence can be started at any time. Each trip but the first one cannot be started before the previous trip in the sequence ends, although it can be delayed as much as needed. Given the duration and the regular cost of each trip in the sequence, can you tell Ástor the minimum cost he must afford so as to complete all the trips in the sequence?

Input

The first line contains an integer N ($1 \leq N \leq 10^4$) representing the number of trips in the sequence. Each of the next N lines describes a trip with two integers D and C ($1 \leq D, C \leq 1000$), indicating respectively the duration (in minutes) and the regular cost of the trip.

Output

Output a single line with a number representing the minimum cost needed to complete all the trips in the order they appear in the input. The result must be output as a rational number with exactly two digits after the decimal point, rounded if necessary.

<p>Sample input 1</p> <pre>2 120 10 10 30</pre>	<p>Sample output 1</p> <pre>40.00</pre>
<p>Sample input 2</p> <pre>3 110 10 10 30 1000 101</pre>	<p>Sample output 2</p> <pre>90.50</pre>
<p>Sample input 3</p> <pre>7 10 1 10 2 10 4 10 4 10 4 10 4 10 1</pre>	<p>Sample output 3</p> <pre>7.00</pre>

Problem D – Database of Clients

Author: Paulo Cezar Pereira Costa, Brasil

Nowadays there are billions of email users. A little-known fact is that some email providers offer way more than the usual `username@provider.com` email address.

Some providers simply ignore dots in usernames. Thus, if John owns the username `johnsmith`, he could tell people that his email address is `johnsmith@provider.com`, `john.smith@provider.com` or `john.s.mith@provider.com`, among others. Emails sent to any of these addresses would end up on his mailbox.

Other providers allow appending the character “+” followed by any combination of letters and/or digits after the username. With this feature, by registering the username `johnsmith`, John would also be able to use `johnsmith+friends@provider.com` and `johnsmith+2x3is6@provider.com`.

Sometimes both features are available at once and in those cases `john.smith+icpc@provider.com` and `john.smith+wants.2.eat.lemon.3.14@provider.com` are valid addresses that John could use.

This is quite useful for users, who can manage different addresses to help organize their mailboxes and easily filter the newsletters eventually sent after registering on a new website. Unfortunately, this also opens up space for abuse. Some websites rely upon the fact that each email address identifies a single user. However, a misbehaving user might easily create multiple accounts by taking advantage of the multiple addresses allowed by the email provider.

After learning all of this your boss got really worried. What if the number of unique users that has been reported to the shareholders is not accurate, bloated by duplicate accounts instead? That brings you to the task at hand: given the list of all email addresses from the users database of the company, you must determine the real number of unique users, assuming that all email providers have both described features available.

Input

The first line contains an integer N ($1 \leq N \leq 1000$) representing the number of email addresses in the database. Each of the next N lines contains a string of at most 100 characters representing an email address in the database. Each email address has the form `localpart@provider` where `localpart` is a non-empty list of labels with a “.” (dot) or a “+” (plus sign) between each pair of consecutive labels, and `provider` is a non-empty list of labels always with a “.” (dot) between each pair of consecutive labels. A label is a non-empty sequence of lowercase letters and/or digits. The character “+” (plus sign) appears at most once in each email address.

Output

Output a single line with an integer indicating the number of unique users in the database.

<p>Sample input 1</p> <pre>2 two.different.providers@now.here two.different.providers@nowhere</pre>	<p>Sample output 1</p> <pre>2</pre>
<p>Sample input 2</p> <pre>2 1.2.3@testing testing@1.2.3</pre>	<p>Sample output 2</p> <pre>2</pre>
<p>Sample input 3</p> <pre>7 alice@e.mail eve@another.mail bob@e.mail joe90@e.mail b.o.b@e.mail bob+new@e.mail bob@another.provider</pre>	<p>Sample output 3</p> <pre>5</pre>

Problem E – Escape, Polygon!

Author: Guilherme A. Pinto, Brasil

A suspicious-looking convex polygon wants to escape its current position by translating itself along some straight-line direction. Three very diligent straight lines want to lock it up by placing themselves along three distinct sides of the polygon. Then, if the triple of lines defines a triangle and the polygon lies inside this triangle, it will be locked up. Otherwise, it will escape.

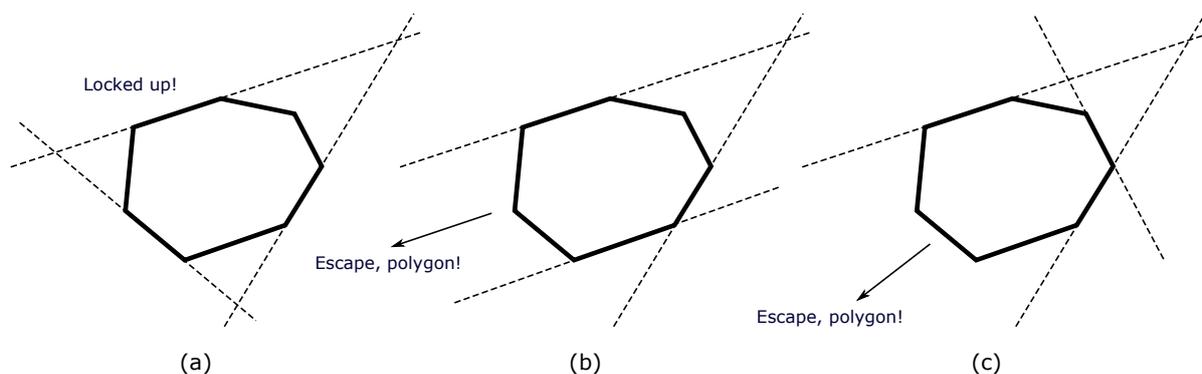


Figure (a) above illustrates a triple that will lock the polygon up. For (b), the lines do not define a triangle since two of them are parallel, and so the polygon will escape. In (c), the polygon lies outside the triangle defined by the triple and it will easily escape.

Given a polygon, you must compute the number of distinct triples of lines that can lock the polygon up.

Input

The first line contains an integer N ($3 \leq N \leq 10^5$) representing the number of vertices of the polygon. Each of the next N lines describes a vertex with two integers X and Y ($-10^8 \leq X, Y \leq 10^8$) indicating the coordinates of the vertex in the XY plane. The vertices are given in counter-clockwise order and they define a simple convex polygon. No three vertices are collinear.

Output

Output a single line with an integer indicating the number of distinct triples of lines that can lock the given polygon up.

<p>Sample input 1</p> <pre>4 0 0 10 0 10 10 0 5</pre>	<p>Sample output 1</p> <pre>1</pre>
<p>Sample input 2</p> <pre>8 0 32 -12 15 -10 -10 0 -12 10 -12 22 0 25 10 18 20</pre>	<p>Sample output 2</p> <pre>18</pre>

Sample input 3 3 10 -10 0 10 -10 -10	Sample output 3 1
Sample input 4 6 -100000000 131 -50000067 -100000000 50000014 -100000000 100000000 -109 70000173 100000000 -90000011 100000000	Sample output 4 6
Sample input 5 4 0 0 10 0 10 10 0 10	Sample output 5 0

Problem F – Fantastic Beasts

Author: Paulo Cezar Pereira Costa, Brasil

The eccentric magizoologist Newt Scamander recently came to Nlogonia to study the fantastic creatures that inhabit this prosperous kingdom. But before he could begin to explore the area an accident disrupted his plans: his suitcase sprang open and his collection of fantastic beasts escaped from the magical object.

The inhabitants of Nlogonia love zoos, and so there are many of them in the kingdom. It turns out that the beasts share Nlogonians' passion for zoos and since the accident they have been visiting the various zoos.

Beasts breaking free and causing trouble is nothing new for Newt so he had trackers put on the beasts since the previous incident. Thus, at any moment he knows the exact position of each of the beasts. After watching the beasts movements for some time he noticed that they follow a peculiar pattern: if a beast is currently in a given zoo, after a unit of time it will either stay in that zoo or it will move to another zoo that depends on the current zoo. All beasts that move to another zoo do this instantly and simultaneously.

With this information Newt conjectured that perhaps it's not so difficult to recover the creatures. He believes that eventually all of them may meet in the same zoo at the same time so he only needs to wait at the right place and capture all the fantastic beasts at once. Given the information Newt has so far, can you help him determine where and when to wait for the beasts? If there are several possibilities, he wants to catch the beasts as early as possible.

Input

The first line contains two integers B ($1 \leq B \leq 10$) and Z ($1 \leq Z \leq 100$), indicating respectively the number of fantastic beasts and the number of zoos. Zoos are identified by distinct integers from 1 to Z . Each of the next B lines describes Newt's findings on a different beast with $Z + 1$ integers P_0, P_1, \dots, P_Z ($1 \leq P_i \leq Z$ for $i = 0, 1, \dots, Z$); the value P_0 is the zoo where the beast initially is, while for $i = 1, 2, \dots, Z$ the value P_i is the zoo where the beast would be after a unit of time if it is currently in the zoo i .

Output

Output a single line with two integers, P and T , indicating that all the beasts will meet for the first time at zoo P after T units of time, or the character "*" (asterisk) if the beasts will never be all at the same zoo.

<p>Sample input 1</p> <pre>2 4 3 4 1 2 3 2 1 1 4 3</pre>	<p>Sample output 1</p> <pre>1 2</pre>
<p>Sample input 2</p> <pre>2 4 3 4 1 2 3 4 1 1 4 3</pre>	<p>Sample output 2</p> <pre>*</pre>

Problem G – Gathering Red-Black Fruits

Author: Mario Silva, Brasil

It's harvest time at Farmer Fred's orchard of red-black trees! But since he's too old to climb trees, Fred brought all his grandchildren to the orchard for a competition of fruit gathering: those who collect the most fruits will be awarded red-black jam jars!

Red-black trees are special, because the same tree gives two different kinds of fruit: the red fruit and the black fruit. That gives Farmer Fred a problem: how to rank children who collected the same amounts of different fruits? For instance, if Abby picked two red and three black fruits, and Bruce picked three red and two black fruits, who should rank higher in the competition? How much should each fruit be worth?

To solve this problem, Farmer Fred decided that each red fruit would be worth r points, and each black fruit would be worth b points, both r and b positive integers. Then he would rank the kids according to the total number of points each one has, ties broken arbitrarily.

All that's left to do now is choose the values of r and b . But Farmer Fred got curious, and now he wants to know in how many different ways he can rank his grandchildren according to the described criteria. Two rankings are considered different if, and only if, there is any child who has different positions in them.

Input

The first line contains an integer N ($2 \leq N \leq 1000$) representing the number of Farmer Fred's grandchildren. Each of the next N lines describes the fruits gathered by a grandchild with two integers R and B ($0 \leq R, B \leq 10^4$), indicating respectively the amounts of red and black fruits the child gathered.

Output

Output a single line with an integer indicating the number of different possible rankings. Print the answer modulo $10^9 + 7$.

<p>Sample input 1</p> <pre>3 0 2 1 2 2 1</pre>	<p>Sample output 1</p> <pre>3</pre>
<p>Sample input 2</p> <pre>4 1 0 1 3 2 2 1 3</pre>	<p>Sample output 2</p> <pre>6</pre>
<p>Sample input 3</p> <pre>4 0 0 3 1 0 0 0 0</pre>	<p>Sample output 3</p> <pre>6</pre>

Problem H – Highway Decommission

Author: Arthur Nascimento, Brasil

Nlogonia’s government is eager to cut down public debt. One of the measures about to take place is the decommission of some highways as most of them incur a high maintenance cost. Each highway connects two different cities and can be traveled in both directions. Using the existing highways it is possible to reach any city from any other city.

Government promises that the impact of the decommission will be minimal in the lives of Nlogonians. In particular they guarantee that after the decommission, for each city the minimum distance needed to travel from that city to the capital of the country will remain the same as it is now, when all the highways can be used.

The Department of Roads of Nlogonia believes that interns are not there just to get coffees or run errands but should do meaningful work instead and that’s why you are assigned the following task. Given the length and maintenance cost of each highway, you must decide which highways will be kept active and which will be decommissioned. As you might guess, the sum of maintenance costs for the remaining highways must be minimum.

Input

The first line contains two integers N ($2 < N \leq 10^4$) and M ($1 \leq M \leq 10^5$), indicating respectively the number of cities and the number of highways. Cities are identified by distinct integers from 1 to N , where city 1 is the capital of Nlogonia. Each of the following M lines describes a highway with four integers A , B , L and C ($1 \leq A, B \leq N$, $A \neq B$ and $1 \leq L, C \leq 10^9$), indicating that there is a highway between cities A and B that has length L and maintenance cost C . Using the existing highways it is possible to reach any city from any other city.

Output

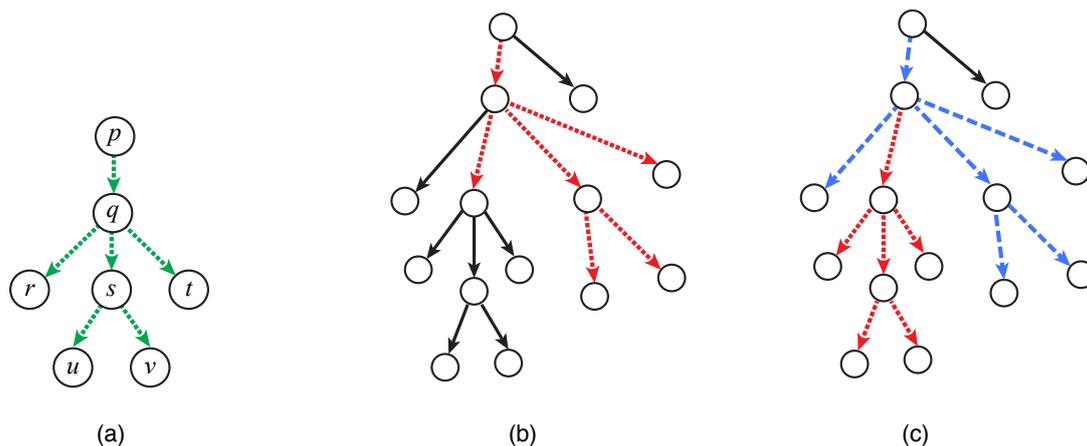
Output a single line with an integer indicating the minimum possible sum of maintenance costs for a set of highways to be kept active. This set of highways must ensure that for each city the minimum distance needed to travel from that city to the capital of Nlogonia remains the same using only those highways.

<p>Sample input 1</p> <pre>3 4 2 3 2 4 2 3 2 2 1 2 5 1 1 3 1 4</pre>	<p>Sample output 1</p> <pre>6</pre>
<p>Sample input 2</p> <pre>2 2 1 2 10 5 2 1 6 11</pre>	<p>Sample output 2</p> <pre>11</pre>

Problem I – Ink Colors

Author: Paulo Cezar Pereira Costa, Brasil

Stick Man left the family tree and went out for adventures. On his journey he found a strange tree with the root on the air and branches directed towards the ground. He decided to paint some of the tree branches to remind himself of home. To do so he wants that branches painted with the same color are all connected and form a stick man. A stick man is a group of six branches (p, q) (q, r) (q, s) (q, t) (s, u) and (s, v) , as show in figure (a) below. Figure (b) shows a tree with one stick man painted and figure (c) shows the same tree with two stick men painted.



Stick Man would like to paint as many stick men on the tree as possible, such that each branch is part of at most a single stick man. Can you help him figure out how many ink colors he needs to buy?

Input

The first line contains an integer N ($1 \leq N \leq 10^5$) indicating the number of nodes in the tree. Nodes are identified by distinct integers from 1 to N , where node 1 is the root of the tree. The second line contains $N - 1$ integers P_2, P_3, \dots, P_N ($1 \leq P_i \leq N$ for $i = 2, 3, \dots, N$), where the value P_i represents that there is a branch (P_i, i) , that is, from node P_i to node i .

Output

Output a single line with an integer indicating the maximum number of stick men that might be simultaneously painted on the tree.

<p>Sample input 1</p> <p>14 1 1 2 2 2 2 5 5 5 6 6 9 9</p>	<p>Sample output 1</p> <p>2</p>
<p>Sample input 2</p> <p>13 13 7 5 1 5 2 5 7 4 2 2 4</p>	<p>Sample output 2</p> <p>2</p>

Problem J – Jeopardized Election

Author: Edwin Niño Velasquez, Colombia

Nlogonian elections are coming up soon and there are many candidates running for President of one of the greatest nations on Earth.

The voting system used in Nlogonia is quite out of the ordinary. Each person votes by making a list of all candidates, in order of preferences of the voter. This means that the first candidate in the list is the one whose proposals please the voter most, and the last candidate in the list is the one whose proposals please the voter least.

Suppose that there are exactly five voters 1, 2, 3, 4 and 5, exactly five candidates A, B, C, D and E, and the voters voted as shown in the following table:

Voter	List of preferences
1	C D A B E
2	B C E D A
3	C E B A D
4	A C B D E
5	D A C E B

To determine the winner, the Electoral Commission first makes a draw, called *Election Ordering*, which contains all the candidates in a certain order. Then each candidate is evaluated following the Election Ordering, until one of them is elected as President. For this to happen, the current evaluated candidate must be the preferred still-running candidate for more than half of the voters.

To make the election system clearer, continuing the example above, suppose that the result of the Election Ordering is C, D, A, E and B. To determine the winner the Electoral Commission would perform the following steps:

- The first candidate evaluated is C. As this candidate is the preferred candidate for just two of the five voters (1 and 3), then C is eliminated.
- Next candidate evaluated is D, who is the preferred still-running candidate for only two voters (1 and 5). Thus, candidate D is also eliminated.
- Candidate A is evaluated next. Since this candidate is the preferred still-running candidate for three of the five voters (1, 4 and 5), candidate A is elected as President and the voting ends.

One of the candidates has managed to corrupt some members of the Electoral Commission, and can now decide what the result of the Election Ordering will be. Also, thanks to various social networks analysis, the candidate knows the list that each voter will vote. The only thing the candidate needs to win the election now is to figure out a proper Election Ordering. As this is not an easy task, someone from the candidate staff anonymously hired you to find an ordering that makes the candidate win. Hurry up, because the draw will occur within the next few hours.

Input

The first line contains two integers C and V ($1 \leq C, V \leq 100$, with V odd), representing respectively the number of candidates and the number of voters. Candidates are identified by distinct non-empty strings of at most 10 uppercase letters. Each of the next V lines describes the vote of a voter, that is, the line contains the list of candidates in order of preference of the voter. All lists contain the same candidates, although candidates may appear in different order. After the votes there is a last line that contains a string W , indicating the candidate that must win.

Output

Output a single line with the Election Ordering that makes candidate W win the election, or the character “*” (asterisk) if it is not possible for W to win. If more than one possible Election Ordering exists, output the lexicographically smallest one.

Sample input 1 5 5 C D A B E B C E D A C E B A D A C B D E D A C E B A	Sample output 1 C B D A E
Sample input 2 3 5 KATE BOB ED BOB ED KATE ED BOB KATE BOB ED KATE KATE BOB ED KATE	Sample output 2 *
Sample input 3 3 5 KATE BOB ED BOB ED KATE ED BOB KATE BOB ED KATE KATE BOB ED ED	Sample output 3 BOB ED KATE

Problem K – KryptoLocker Ate my Homework

Author: Arthur Nascimento, Brasil

Your friend’s computer was infected by KryptoLocker last night. KryptoLocker is a ransomware that encrypts user’s data making it unusable until a ransom is paid to the person behind the attack.

Luckily your friend noticed there was something weird happening and was able to stop the malicious process before all data was lost. Unfortunately it didn’t happen in time to save the homework he had just finished.

“*The dog ate my homework*” never really worked as an excuse for most teachers and probably things won’t be different with “*KryptoLocker encrypted my homework*”.

But not all hope is lost. For this homework each student was assigned an array of integer numbers $v_1 \leq v_2 \leq \dots \leq v_N$. Then, each student had to generate a list of 2^N numbers, containing the sums of the elements in each possible subsequence of the array. KryptoLocker only corrupted the first line of the file containing your friend’s answer and this was the line where the array was written. So, all sums are still known and that should be enough to recover the array.

Can you help your friend save his homework?

Input

The first line contains an integer N ($1 \leq N \leq 18$) indicating the number of elements in the array assigned to your friend. Each of the next 2^N lines contains an integer S ($-10^9 \leq S \leq 10^9$) representing the sum of the elements in a subsequence of the array. The sums of all possible subsequences appear in the input, in no particular order. There is at least one array whose sums are the values given.

Output

Output each different array that may have been assigned to your friend, one array per line, with the lines lexicographically sorted. Two arrays are considered different if, and only if, they contain different elements in at least one position.

<p>Sample input 1</p> <pre>3 9 8 7 0 15 16 17 24</pre>	<p>Sample output 1</p> <pre>7 8 9</pre>
<p>Sample input 2</p> <pre>2 1 0 -1 0</pre>	<p>Sample output 2</p> <pre>-1 1</pre>

Problem L – Looking for the Risk Factor

Author: Juan Pablo Marín Rosas, México

For testing a new cryptographic algorithm, engineers working for a large investment bank need to compute a value they named the *Risk Factor* of the algorithm. Informally, the Risk Factor is the amount of numbers less than or equal to a certain value N , that aren't multiples of prime numbers greater than a certain value K .

More formally, given the values N and K , the Risk Factor is the number of elements of the following set:

$$\{x \text{ such that } 2 \leq x \leq N \text{ and for every prime divisor } p \text{ of } x, p \leq K\}$$

The engineers need to compute the Risk Factor for different values of N and K and have prepared a set of queries for you to answer. Can you help them?

Input

The first line contains an integer Q ($1 \leq Q \leq 5 \times 10^4$) representing the number of queries that the engineers prepared for you. Each of the following Q lines describes a query with two integers N and K ($2 \leq N, K \leq 10^5$).

Output

Output Q lines, each line with an integer indicating the Risk Factor for the corresponding query of the input.

Sample input 1	Sample output 1
4	6
10 3	6
10 4	7
15 3	4
5 20	

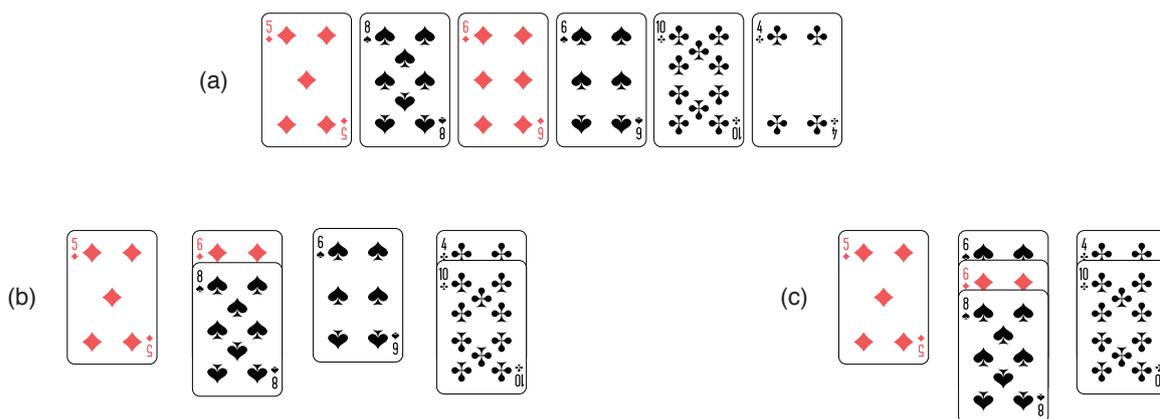
Problem M – Mount Marathon

Author: Inés Kereki, Uruguay

Mount Marathon is a solitaire game that is played using a regular deck of 52 cards. To start the game the player shuffles the deck and lays N cards face up on the table, forming a straight line of N piles, each pile having a single card. No other cards are used during the rest of the play. Then the player repeatedly moves a pile on top of another pile until no more movements are available. The goal of the game is to end up with the minimum number of piles. When moving a pile p on top of another pile q , the following conditions must hold:

- Pile p must be a single-card pile.
- The value of the only card in pile p must be greater than or equal the value of the card that is on top of pile q .
- Pile q must be the next pile remaining immediately on the right of pile p .

Figure (a) below shows a configuration with six cards at the beginning of the game. The player may move the fifth pile on top of the sixth pile, and then the second pile on top of the third pile; since no more movements are available, this would conclude the game with four piles remaining, as it can be seen in figure (b). However, in this case it is possible to end up the game with just the three piles that appear in figure (c).



Given the initial piles, you must determine the minimum number of piles that it is possible to obtain at the end of the game.

Input

The first line contains an integer N ($1 \leq N \leq 52$) representing the number of cards in the game. The second line contains N integers C_1, C_2, \dots, C_N ($1 \leq C_i \leq 13$ for $i = 1, 2, \dots, N$) indicating the values of the cards in the initial piles, from left to right. Each card value appears at most four times.

Output

Output a single line with an integer indicating the minimum number of piles that it is possible to obtain at the end of the game.

<p>Sample input 1</p> <p>6 5 8 6 6 10 4</p>	<p>Sample output 1</p> <p>3</p>
<p>Sample input 2</p> <p>1 13</p>	<p>Sample output 2</p> <p>1</p>

Sample input 3 5 2 4 6 8 10	Sample output 3 5
Sample input 4 11 13 1 1 1 13 7 8 10 4 2 1	Sample output 4 4