Problem A. 149576. Seems easy but not.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You're given a text where you need to count how many times each word appears. Output the list of words with their corresponding counts. The list must be sorted by the count in descending order. If two counts are equal, sort lexicographically by the word.

Input

Single line — string $Text$ with a length of $(1 \leq N \leq 100)$.

Output

Output the count of words from the text. The resulting list must be sorted from max to min count. If some words have the same count of appearances, then sort them in lexicographical order.

Examples

standard input	standard output

`some repeating words words words but`	`words : 3`
	`but : 1`
	`repeating : 1`
	`some : 1`

`how do how do do you doing how you how`	`how : 4`
	`do : 3`
	`you : 2`
	`doing : 1`

`illl`	`illl : 1`

Note

Pay attention to the format of the output - print spaces before and after the `":"` sign as it is shown in the example.

You can use a `map` to count the words, then put all the elements into a `vector` of `pairs` and sort it by writing a comparator for the `sort` function.

Problem B. 76644. Bad WI-FI connection

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Unfortunately, when one of the assistants has been preparing midterm marks, the WIFI connection was cut off. After reconnection, he realized that a list of all students and their marks is mixed up. He knows that students were in alphabetical order and marks were in ascending order. Help the assistant to restore the list.

Input

The first line contains a single integer — N, the amount of students. Next N lines will be given pairs of student name(string) and mark(int).

Output

Print the restored list. Each student should be in a separate line.

Examples

standard input	standard output
5 Azamat 5 Eldar 15 Temirlan 30 Assyl 25 Yerdaulet 15	Assyl 5 Azamat 15 Eldar 15 Temirlan 25 Yerdaulet 30
10 Dina 40 Bagdat 35 Aldiyar 25 Assyl 15 Dauren 30 Rayan 20 Aldikk 35 Dauzee 40 Askar 45 Derbes 15	Aldikk 15 Aldiyar 15 Askar 20 Assyl 25 Bagdat 30 Dauren 35 Dauzee 35 Derbes 40 Dina 40 Rayan 45

Note

Use the sort function from `<algorithm>`. Read about it on ejudge reference or cplusplus.com. You should note that the first letter of each name is uppercase.

Problem C. 77392. Generate power

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Most of the mathematicians say that the function `f(x) = pow(x, x)` has an interesting behavior. However, it is hard to calculate such a function for the value n; that is why you should help them to create a sequence of elements n power n, where n is the index of the element in the range $(0, N]$. Make mathematicians' lives easier.

Input

Input will contain only one integer N, where N is the last index of the sequence. Indexes start from 0.

Output

Create a sequence of n power n, separating elements by a single empty space.

Examples

standard input	standard output
5	1 1 4 27 256 3125
4	1 1 4 27 256

Note

You need to write a function that will find n power n and use it via the function `generate()` from `<algorithm>`. You may use `long long` to save the generated number.

Use the function `powl()` instead of `pow()` to calculate n power n.

Hint: you can use a global variable for the index and increment it by 1 inside the `generate()`.

Problem D. 77686. Squack

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Bagdat does not like two-digit squares in a sequence. She would like to erase all of them. Help Bagdat do it.

Input

In the first line, you are given an integer n ($1 \leq n \leq 10^{60}$).

Output

Print the reversed number n after the required operations. If the resulting sequence is empty, just print `Stack is empty`.

Examples

standard input	standard output
161496	Stack is empty
110256	6011

Note

Even `long long` type is not enough for this task. You need to use `string` to save the value of the number.

Use `stack` to solve this problem.

Hint: the type of elements in the `stack` is `char`.

Problem E. 77721. Hard-sort

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 megabytes

We hope you have learned from previous tasks how to use `sort()` and write a comparator function. Now, you have a task to sort a non-static 2D-array. Sizes of rows are undefined. Sort all of the parameters in non-decreasing order. First of all, sort rows by the sum of all elements in the row; if they are equal, sort by the size of the row; if they are equal, sort by the corresponding elements (which row has a larger corresponding element is considered "greater"); and if they are also equal, it means that they are in the right order.

Input

The first line contains a single integer N, which is the number of rows. The next N lines first contain a number M, which is the size of the row, followed by M numbers separated by a single empty space.

Output

Print a sorted "non-static 2D-array". Print each given row in its own line. Separate elements of the row by a single empty space.

Example

standard input	standard output
5 4 9 8 7 6 4 5 4 3 2 4 1 1 1 1 4 1 7 6 7 4 5 4 7 8	1 1 1 1 5 4 3 2 1 7 6 7 5 4 7 8 9 8 7 6

Note

Write a comparator to implement specified sorting.

Hint: you should use a vector that contains vectors. The comparator will take 2 vectors of integers as parameters.

Problem F. 77723. Where are the students from?

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Every year, at summer time, a selection committee takes documents of students from different places of KZ. They want to know the percentage of students from each place. Help them calculate it.

Input

The first line provides a number N ($1 \leq N \leq 40$) — the amount of days that the committee worked. Then you get N days in the input, where each day consists of:

- A row with number M ($1 \leq M \leq 300$) — how many places are sending documents this day.
- M rows contain the name of the place and the amount of students from that place, separated by a space.

Output

Each printed row should contain the name of the place and the percentage of students from that place, separated by a single empty space. Sort the places alphabetically.

Example

standard input	standard output
2 15 Almaty 80 Kostanai 243 Oskemen 243 Oral 219 Turkistan 93 Karagandy 159 Aktobe 36 Kostanai 136 Oskemen 116 Aktau 92 Turkistan 26 Taraz 196 PetroPavlsk 62 Aktobe 56 Aktobe 95 2 Astana 112 Pavlodar 2	Aktau 4.67955 Aktobe 9.5117 Almaty 4.06918 Astana 5.69685 Karagandy 8.08749 Kostanai 19.2777 Oral 11.1394 Oskemen 18.2604 Pavlodar 0.101729 PetroPavlsk 3.15361 Taraz 9.96948 Turkistan 6.0529

Problem G. 77731. GPA

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Unfortunetely, after finals our uninets was damaged again. But one of the students managed to write points for the first and second attestation and final, amount of credit. Help to calculate student's GPA.

Input

The first line contains integer n, which amount of subjects that were taken in previous semester. The next N lines contain four integers: 1st attestation, 2nd attestation, final and credits.(separated by space).

Output

Print single double number - GPA.

Examples

standard input	standard output
6 26 28 8 4 17 12 36 4 25 27 6 3 13 16 15 2 30 15 30 4 6 11 10 4	0.507937
5 30 21 22 4 23 26 8 2 21 12 12 4 9 9 16 1 22 5 40 2	0.717949

Note

$$GPA = \frac{\sum_{i=1}^{n} (GP_i \cdot \text{Credits}_i)}{\sum_{i=1}^{n} \text{Credits}_i}$$

where:

- n is the number of subjects
- GP_i is the grade point for the $subject_i$
- $Credits_i$ is the credit hours for $subject_i$

Alternatively, simplified:

For each subject, multiply the Grade Point (GP) by the amount of credits. Sum it all up and divide by the total amount of credits.

If either the sum of 1st and 2nd attestation is less than 30, or the final exam mark is less than 20, GP for that subject is considered 0.

Beginning from 50 points, GP is increasing by 1/3 with each 5 points for the individual subject. For example: 50-54 points = 1.0 GP, 55-59 points = 1.0 + 1/3 GP, 60-64 points = 1.0 + 2/3 GP, etc. 95+ points = 4.0 GP.

Problem H. 77736. Competition

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

A few days ago a hackathon was held at KBTU. In this hackathon, 2 students could participate as a team. Jury marked each student's name individually and then later they got the sum of their scores. But there were some issues after the competition ended. All of the students had written only theirs nickname at the registration time. Moreover, some students have registered twice (or maybe more). Jury decided: if 2 teams of students have the same names scores - they are same team, that registered two or more times. But if they have different scores they are a different team. Shortly, for 2 teams to be the same their information and score has to be completely the same. Help the Jury create a list of teams with their scores without any duplicates.

Input

In the first line, an integer N is given — amount of registered teams. Next N lines contain information about the team in the following pattern:
`student-1 score-1 student-2 score-2`.

Output

Print the list of unique teams (according to the decision of the jury), each team in a separate line. Each line should contain names of the first and the second participants and their total score.

Examples

standard input	standard output
2 Assyl 40 Azamat 40 Assyl 40 Azamat 50	Assyl and Azamat 80 Assyl and Azamat 90
2 Assyl 40 Azamat 40 Assyl 40 Azamat 40	Assyl and Azamat 80

Note

Use only `map` and `pair`. No `vector`, `set` or other STL containers.

Problem I. 76658. Next permutation

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Askar likes course of discrete structures. Now, he is trying to solve given homework that is about next permutation. In the task was said that he needs find all possible next permutation of the given sequence. Some of numbers in the sequence were duplicated. Help Askar to solve homework.

Input

In the first line given single number N - amount of numbers in sequence. The second line contains of N numbers.

Output

Print all possible next permutation of the sequence. Elements should be separated by single empty space. Print each of permutaion in own row.

Examples

standard input	standard output
3 1 2 3	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1
10 10 10 10 10 10 10 10 10 10 10	10
6 6 5 4 3 2 1	6 5 4 3 2 1

Note

Use function for next permutations in algortihm. Read about it on cplusplus.com

Problem J. 76967. Instead of ...

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

It is obvious that most SITE students like to solve interesting problems. Aldiyar explained to Assyl a statement of one of such problems. He wrote a sequence of numbers on a piece of paper and showed it via a mirror. As you can guess, Assyl sees the reversed sequence. Give hints for Assyl which positions the numbers originally occupied.

Input

In the first row, a single number N is given - an amount of numbers in the sequence.
The next line contains N numbers of the sequence.

Output

Print hints as in the examples. If 2 numbers in the corresponding positions are equal, then print `OK` for that case.

Examples

standard input	standard output
5 5 4 3 2 1	Instead of 5 here was 1 Instead of 4 here was 2 OK Instead of 2 here was 4 Instead of 1 here was 5
3 2 1 3	Instead of 2 here was 3 OK Instead of 3 here was 2
5 1 2 3 2 1	OK OK OK OK OK

Note

Just simply use the function `reverse` from `<algorithm>`

Problem K. 77054. Instead of ... 2

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

I hope previous problem was easy for you. If previouly sequence was just reversed. Right now, sequence is firstly rotated then reversed. Restore initial sequence again.

Input

In the first line given numbers N and M point of rotation. Second line contains of N numbers.

Output

Print original sequence in one row separated by single empty space.

Example

standard input	standard output
5 2 2 1 5 4 3	1 2 3 4 5

Note

Use function reverse and rotate in <algorithm>. Read on http://www.cplusplus.com/reference/algorithm/rotate/?kw

Problem L. 77065. New Rules

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

From course of math we know that the number is called prime if and only if number is greater that 1 and divisible by 1 and itself. But one of the students tried to apply this rule for negatives and zero. He thinks, if positive of number is prime, number is prime too. Zero is not prime. You task is to calculate all primes by new rules.

Input

In the first row given single number N amount of numbers. Second line contains sequence of integers.

Output

Find how many primes are in sequence.

Examples

standard input	standard output
8 3 4 5 7 9 -9 -7 0	4
4 17 -17 1 3	3

Note

Write function to find prime numbers, you have to use function count if in "algorithm".

Problem M. 77327. Arithmetic progression in a nutshell

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Dina loves her young brother. She wants to teach him arithmetic progression. To show what is the arithmetic progression and it's sum, she decided to write sequence like 1 2 2 3 3 3. Help Dina to teach him.

Input

Input contains only one single integer N(the last number of the sequence).

Output

Create required sequence. All members should be in the same row separated by single empty space.

Examples

standard input	standard output
5	1 2 2 3 3 3 4 4 4 4 5 5 5 5 5
4	1 2 2 3 3 3 4 4 4 4

Note

Use vector for this task. Think about size of the vector(what it might be?).You have to use function fill() in <algorithm>.

Problem N. 77330. unique x set

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

We know that it is easy to find all unique elements in sorted order via set. But also, there are exist such functions as unique that will give iterator without consecutive duplicates as a result. In this task you will be given two sequences. Erase all consequent duplicates for both vectors. Then merge them to the new vector(Turn by turn,first element will be first of first, second element of the new vector will be first of the second). Do the same thing with a new vector(Use unique again).

Input

The first line will contain numbers N and M.(Number of elements of the corresponded vectors). Next two lines contain sequences of integers.

Output

Print sequence separated by single empty space.

Examples

standard input	standard output
5 4 5 5 4 3 2 4 5 4 3	5 4 5 3 4 2 3
1 2 1 1 1	1

Note

You have to use only vector and function unique() in <algorithm>. Read about it on cplusplus.com. It is guaranteed that difference between size of two "unique"vectors no more than 1.

Problem O. 77351. Possible palindromes

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Aibek likes palindromes. He always gets any sequence and try make palindrome. If it is possible to make many palindromes, prefers first in "ascending"order(Even when sequence is already palindrome). Help Aibek to create palindrome.

Input

In the first line you are given single number N, which number of elements. Second line contains N numbers.

Output

Print first possible palindrome.If it is not possible, print "Impossible".

Examples

standard input	standard output
5 3 1 2 1 3	1 3 2 3 1
3 1 2 3	Impossible
4 2 1 1 2	1 2 2 1

Note

You have to write function that will check is the sequence is palindrome. Use function in <algorithm> that will find all combination of the sequence(previously you used it). Moreover, in <algorithm> exist 2 such functions. You may use both, but methods will differ only in using sort.

Problem P. 77388. Binary for each

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

All of computer in the world use binary system for any operation. Let try to simulate it. Represent each elements of given vector as binary.

Input

In the first line you are given single number N, where N is amount of elements in the vector. The second line contains of N elements.

Output

Print binary representation for each element of vector in own row.

Example

standard input	standard output
5 1 2 3 4 5	1 10 11 100 101

Note

You have to write function that will convert number to binary and you must use function foreach in "algorithm". Think about to use take you function as argument for foreach.

Problem Q. 195978. Boris and the beads.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Boris decided to make a beautiful piece of bead jewelry for his mother. He has a sequence of different beads, but he wants the decoration not to have the same beads going in a row. Help him write a program that will remove identical beads in a row and leave only one of them.

Input

First line contains integer N, $(1 \leq N \leq 10^4)$ - size of the sequence.

Second line contains space-separated N integers elements of array A - beads, where $(1 \leq A[i] \leq 10^3)$

Output

Print space-separated elements of the new sequence.

Examples

standard input	standard output
7 10 10 20 30 30 20 20	10 20 30 20
5 2 2 1 1 5	2 1 5

Note

Hint : Use function *unique*() from the `<algorithm>` library.

Problem R. 192812. Eve and Permutation.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Eva came up with a system of permutations with fixed points. Its essence is $P[1..N]$ of size N is a set of numbers from 1 to N arranged in a certain order. At the same time, each of these numbers must be present exactly once in it - without duplicates. There are also so-called fixed points. A number i is called a fixed point for a permutation P if $P[i] = i$. For example, in a permutation $1, 3, 4, 2, 5$ there are exactly two fixed points: 1 and 5, and the permutation $5, 4, 3, 2, 1$ has no fixed points.
Two numbers are given: N and K. Find the number of permutations of size N with exactly K fixed points.

Input

Input data is two integers separated by a space N and K,where $(1 \leq N \leq 9)$ and $(0 \leq K \leq N)$.

Output

Output the number of permutations.

Examples

standard input	standard output
4 0	9
1 1	1
5 0	44
4 0	9
8 1	14832
2 1	0
6 2	135
1 0	0
9 6	168

Problem S. 194148. Brute force.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You are given array of numbers which used in password, you have to create algorithm of brute force to hack digital lock. Array of numbers may contain duplicates because of some error while getting password. You should avoid duplicated number and create list of possible pass codes.

Input

The first line contains integers N, where $1 \leq N \leq 9$, second line contains array A of size N, where $0 \leq A[i] \leq 8$

Output

Print on separate lines possible passwords where password is digits separated by space.

Examples

standard input	standard output
4 1 1 2 3	1 2 3 1 3 2 2 1 3 2 3 1 3 1 2 3 2 1
1 1	1
5 6 2 2 8 7	2 6 7 8 2 6 8 7 2 7 6 8 2 7 8 6 2 8 6 7 2 8 7 6 6 2 7 8 6 2 8 7 6 7 2 8 6 7 8 2 6 8 2 7 6 8 7 2 7 2 6 8 7 2 8 6 7 6 2 8 7 6 8 2 7 8 2 6 7 8 6 2 8 2 6 7 8 2 7 6 8 6 2 7 8 6 7 2 8 7 2 6 8 7 6 2

Problem T. 105689. Merge.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Given two sorted integer arrays a and b, merge a into b as one sorted array.

Input

The first line contains one integer n ($1 \leqslant n \leqslant 10^3$) — a size. The second line contains n integers a_i ($1 \leqslant a_i \leqslant 10^3$) — elements of array. The third line contains one integer m ($1 \leqslant m \leqslant 10^3$) — b size. The next line contains m integers b_i ($1 \leqslant b_i \leqslant 10^3$) — elements of array.

Output

Array B into sorted order.

Examples

standard input	standard output
5 4 5 6 7 10 5 2 4 6 7 8	2 4 4 5 6 6 7 7 8 10
3 1 4 7 1 6	1 4 6 7
4 2 5 7 8 3 2 4 9	2 2 4 5 7 8 9
2 3 10 4 1 3 6 8	1 3 3 6 8 10
1 4 4 1 1 5 6	1 1 4 5 6

Problem U. Get sorted array

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Darkhan has a sequence p = {p_1, p_2, ... , p_n} which is permutation of {1, 2, ... , n}. He can perform the following operation at most once: choose integers i and j (1 <= i, j <= n) and swap p_i and p_j. Help him to get sorted array or tell him that it is impossible.

Input

First line contains one integer n (1 <= n <= 50). Second line contain n integers p_i (1 <= p_i <= n).

Output

Print "Yes"if it is possible to get sorted array, otherwise print "No".

Examples

standard input	standard output
5 5 2 3 4 1	YES
5 2 4 3 5 1	NO
7 1 2 3 4 5 6 7	YES

Note

You can also choose not to perform operation.

Problem V. 149504. The Da Vinci Code — anagram.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

We continue our journey through the amazing book — The Da Vinci Code. This time, Sophie Neva needs to unravel the word hidden under the anagram. To do this, she wants to find out all the variants of anagrams for a given word.

An anagram is a word formed by rearranging the letters of a different word or phrase, typically using all the original letters exactly once. For example, the word *binary* can be rearranged into *brainy*, and also the word *adobe* into *abode*.

Input

You are given a string s — initial word ($3 \leq s.size() \leq 10$).

Output

Write a function for printing all the variants of the anagram for the given word in lexicographic order, each separately on a new line.

Example

standard input	standard output
hello	The anagram variants for string hello are: ehllo ehlol eholl elhlo elhol ellho elloh elohl elolh eohll eolhl eollh hello helol heoll hlelo hleol hlleo hlloe hloel hlole hoell holel holle lehlo lehol lelho leloh leohl leolh lhelo lheol lhleo lhloe lhoel lhole lleho lleoh llheo llhoe lloeh llohe loehl loelh lohel lohle loleh lolhe oehll oelhl oellh ohell ohlel ohlle

Note

Hint: use the $< algorithm >$ library and the function $next\_permutation()$

Problem W. ZA WARUDO TOKI WO TOMARE

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

DIO is fighting with JOJO. DIO wants to cast time stop, but for this DIO needs to find at least one palindrome in given string `s` by permutations of letters, help DIO, is he can cast ZA WARUDO TOKI WO TOMARE.

Input

You are given single string `s`.

Output

Print ZA WARUDO TOKI WO TOMARE, if given string could be palindrome permutation, otherwise print JOJO

Examples

standard input	standard output
jojo	ZA WARUDO TOKI WO TOMARE
jojorefer	ZA WARUDO TOKI WO TOMARE
aabc	JOJO
asdasd	ZA WARUDO TOKI WO TOMARE

Note

jojo possible palindrome -> jooj

jojorefer possible palindrome -> ejorfroje

Problem X. 194308. Designer.

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You are working as Product Designer and you have to generate logo name for company using special symbols that company presented to you. Generate all possible strings to select one after process, but to make that without script code will be harder. So you have to write a code.

Input

The first line contains string where string size N will be in range $1 \leq N \leq 9$ with symbols in range: a-z, A-Z and 0-9.

Output

Output all possible combinations without space in lexicographic decreasing order.

Examples

standard input	standard output
AB	BAAB
Y1a	aY1a1YYa1Y1a1aY1Ya
AAbb	bbAAbAbAbAAbAbbAAbAbAAbb

Problem Y. 73771.Reverse the string

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

You're given string s, you need to reverse this string.

Input

In first line given string s.

Output

Output answer for the problem.

Examples

standard input	standard output
abcde	edcba
abba	abba

Problem Z. Daniil Watch

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes

Daniel was given a very expensive rolex watch for his birthday. But it turned out to be defective. Now the watch needs repair, you need to sort the time as shown below. the first column is hours, then minutes and seconds.

Input

Integer n, (1<n<100).

Output

print answer.

Examples

standard input	standard output
3 18 6 37 8 47 35 22 55 33	8 47 35 18 6 37 22 55 33
5 18 6 37 8 47 35 22 55 33 18 20 37 7 39 6	7 39 6 8 47 35 18 6 37 18 20 37 22 55 33

Note

You cannot use the sort function and You must definitely solve this problem using the function, otherwise the problem will be counted as 0 points.