Problem A. Ministry of Truth

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

In whiteblack on blackwhite is written the utterance that has been censored by the Ministry of Truth. Its author has already disappeared along with his whole history, and now, while Big Brother is watching somebody else, you, as an ordinary official of the Minitrue, have to delete some letters from the utterance so that another utterance will appear, which has been approved of by the Ministry.

The Ministry of Truth defines a *word* as a nonempty sequence of English letters and an *utterance* as a sequence of one or more words separated with one or more spaces. There can also be spaces before the first word and after the last word of an utterance. In order to compare two utterances, one should delete all the leading and trailing spaces and replace each block of consecutive spaces with one space. If the resulting strings coincide, then the utterances are considered to be equal.

Input

The first line contains the original utterance and the second line contains the utterance that must be obtained. The length of each utterance is at most 100 000 symbols. The words in both utterances are separated with exactly one space; there are no leading and trailing spaces in each line. The original and the required utterances are different.

Output

If you can't carry out your order, output «`I HAVE FAILED!!!`» in the only line. Otherwise, output the original utterance replacing the letters that are to be deleted with the underscore character.

Examples

input.txt	output.txt
`Preved to Medved` `Preved Me`	`Preved __ Me____`
`this is impossible` `im possible`	`I HAVE FAILED!!!`

Problem B. Forgotten Technology

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

Winter. End of December. Snowing. The shrill wind catches snowflakes and carries them among the stone buildings. The monastery looks deserted — doors are blocked by snow drifts, shutters are closed. Only the windows of the scriptorium — the largest ones in the monastery — are wide open. Dim winter light illuminates the room. A few monks sit there in complete silence. The room is so cold that monks' fingers go numb and white, but they continue to quickly write with quill pens, trying to do as much as possible during the short December day.

A sacred rite is being performed in the scriptorium — information is being copied.

One scribe, not long ago promoted from an apprentice, is writing his first book. His lips are moving, he mumbles every word he's copying to ensure against mistakes. Another one has copied dozens of books, every letter he writes is indistinguishable from the original. He devoted all his life to writing, and he has no other skills.

A monk, called corrector, is sitting in the adjacent room. His duty is no less important — he's leaving notes and corrections in the margins of a book. Also, binders and illuminators are working in the scriptorium. They add final touches to a book before it ends up in rich layman's hands. Hard work of all these monks is essential for survival of the monastery.

But daylight fades, scribes put quills aside and go away to their cells. The day is drawing to a close. The year 1439 is drawing to a close. The epoch of monk scribes is drawing to a close. Next year, 1440, Johannes Gutenberg will print his first book.

***

Autumn. Mid-November. Raining. Young woman is sitting at the diner on the corner of Broadway and 112th Street. She looks out the window at the dark clouds hanging over the city. Maybe it's just the weather, but all the morning she feels strange emptiness in her soul. She feels like apathetic observer, like nothing concerns her.

The waiter comes to her and fills her cup with coffee halfway. She tries to argue, but he doesn't look at her anymore. He's looking at a woman entering the diner and shaking her umbrella. He smiles and greets her. Young woman turns her head away from them and pours milk in her coffee.

Feeling bored, she picks the newspaper. Nothing interesting in it, except a story about an actor who drank too much, slipped in his apartment and died. However, his name doesn't sound familiar to her. She turns the pages looking for the horoscope and comic strips.

She thinks someone is watching her. She turns her head and sees a woman behind the window, looking inside. In a moment, she realizes that the woman is just looking at her reflection, fixing her stockings.

And the rain goes on and on. She hears the bells of the Cathedral of Saint John the Divine, located opposite the end of the street. The bells ringing remind her of an old friend, with whom she once had a little picnic on the steps of the cathedral at midnight.

She finishes her coffee and leaves the diner. The diner's name is «Tom's». Young woman's name is Suzanne Vega. In a few months she'll write a song about this morning. And in a decade audio engineer from Germany will use the song to fine-tune his compression algorithm. He'll listen to it thousands of times, before final version of the compression scheme known as MP3 will be ready.

***

In the Museum of Science History, in the «Forgotten Technology» section, there is an exhibit called «Mass Copy Victims». Under the glass, a 15th century hand-written book lies on the velvet cloth. A 20th century compact disk lies on top of the book. These technologies have been forgotten with the advent of new ways to copy and distribute information. Hand-written books lost competition with cheap output of printing presses, while compact disks were forced out by pervasive digital networks. And now you can find them only in private collections and this museum.

Also, a little mouse lives in the museum. She lives there because she's curious and adores museums. She gnawed through the bottom of the showcase, and now she wants to bite a bit of the disk and take it away to her hole. But she's afraid that the alarm will set off before she makes it. Help her to find the shortest path to the disk.

The book is a rectangle. The disk is lying on the book, so its center is either inside the rectangle or on its border. The mouse can run only on the cloth, her own dimensions are negligible. In her initial position, there is a non-zero distance between her and the book/the disk. To bite the disk, the mouse has to run to a point where the disk hangs over her, that is the point should be strictly outside the rectangle.

Input

The first line contains the coordinates of the vertices of the rectangle, listed in the counter-clockwise order. The next line contains the coordinates of the center of the disc and its radius. The third line contains the coordinates of the initial position of the mouse. All numbers are integers and don't exceed 1 000 in absolute value. The radius of the disc is equal or greater than 1.

Output

Output the length of the shortest path to the disc, accurate to at least 10^{-5}. It is guaranteed that the required path exists.

Example

input.txt	output.txt
-4 -3 8 6 5 10 -7 1 1 7 5 7 -1	12.07107

Problem C. Endgame Database

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

For many years programmer Starostin had been writing a checkers-playing program for the $n \times n$ board. The triumph moment was very near — he was going to issue the final version soon and enjoy the glory of the creator of the best checkers program in the world. The program was almost invincible already, and with the new big endgame database there would be no equals to it. It only remained to generate that database. . .

When he finished writing the generator of moves in assembler, Starostin found out that there was no empty space left on his hard disk. The endgame database would require a huge amount of memory. In order to estimate the number of hard disks he would have to buy, Starostin decided to compute in advance the number of positions in his database.

The database must contain only those positions in which there are exactly k pieces on the board. Each piece is characterized by its color (white or black) and type (a man or a king). There must be at least one piece of each color on the board, all pieces must occupy black squares, and men mustn't stand on their crowning squares (this means that white men can't be in the last rank and black men can't be in the first rank).

Input

The only input line contains space-separated integers n and k ($4 \leqslant n \leqslant 1\,000$; $2 \leqslant k \leqslant \frac{n^2}{4}$; n is even).

Output

Output the number of positions in the endgame database computed modulo $10^9 + 7$.

Example

input.txt	output.txt
4 2	172

Problem D. Theft of the Century

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

An evil-doer got into a depository where gold and platinum bars were stored and took out n sacks with bars. In each sack there were k bars and all the bars in each sack were made of the same metal. To arouse less suspicions, the criminal painted all the bars black, which made gold bars and platinum bars look the same.

Soon he found a buyer for the platinum bars. However, it turned out that the thief didn't know which bars were in which sack. Still, he kept his head and decided to find that out using a balance that could show exact weights. The thief knew that the mass of a gold bar was x kilograms and the mass of a platinum bar was y kilograms. Help the thief determine in one weighing in which sacks there are gold bars and in which sacks there are platinum bars. Tell him how many bars from each sack he should put on the balance in order to determine which sacks contain platinum bars.

Input

The only input line contains space-separated integers n, k, x, and y ($3 \leqslant n \leqslant 20$; $1 \leqslant k \leqslant 10^6$; $1 \leqslant x < y \leqslant 10$).

Output

If it is possible to determine in which sacks there are platinum bars in one weighing, output «`YES`» in the first line and output n space-separated integers in the second line. These numbers must describe how many bars from each sack should be put on the balance. If there are several solutions, output any of them. If there is no solution, output «`NO`» in the only line.

Examples

input.txt	output.txt
`4 8 9 10`	`YES` `4 5 6 8`
`3 3 9 10`	`NO`

Problem E. Chinese Hockey

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

Sergey and Denis closely followed the Chinese Football Championship, which has just come to an end. They supported the *Katraps* and *Komolotiv* teams, but, unfortunately, these teams tied for last place in the championship. Sergey was so disappointed that he suggested Denis that they change to hockey fans.

There are n teams competing in the Chinese Ice Hockey Championship. During the season, each team must play with each other team exactly one game. If a team wins in the regulation time, it gets 3 points and the losing team gets 0 points. If the regulation time is ended in a draw, then the overtime is played. The team that wins in the overtime gets 2 points and the team that loses gets 1 point. A game can't end in a draw in ice hockey.

Denis wants to determine which team he will support. In order to make the choice, he has found a table on the Web in which it is shown for each team how many points it scored in the last year's season. Sergey suspects that there is a mistake in this table because no all-play-all tournament could end with such results. Is Sergey right?

Input

The first line contains an integer n ($2 \leqslant n \leqslant 200$). The second line contains n space-separated non-negative integers; they are the scores of the teams in the previous championship. The scores are given in the non-increasing order. The sum of all the scores is $\frac{3n \cdot (n-1)}{2}$. None of the teams scored more than $3 \cdot (n-1)$ points.

Output

If Sergey is right and there is a mistake in the table, output «`INCORRECT`» in the only line. Otherwise, in the first line output «`CORRECT`» and in the following $\frac{n \cdot (n-1)}{2}$ lines output the results of the games. Each result must have the form «`i ? j`», where i and j are the numbers of the teams that played the game and `?` can be `<`, `<=`, `>=`, or `>`, which means that the first team lost in the regulation time, lost in the overtime, won in the overtime, and won in the regulation time, respectively. The teams are numbered from 1 to n in the order in which they are given in the input.

Examples

input.txt	output.txt
4 8 7 2 1	CORRECT 2 <= 1 3 >= 4 1 > 3 4 < 2 1 > 4 2 > 3
4 8 8 1 1	INCORRECT

Problem F. Mnemonics and Palindromes 3

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

As you remember, when Vasechkin was preparing a problem for the latest student contest, he spent a lot of time trying to invent an unusual and complex name for this problem. The name that Vasechkin had invented was so complex that none of the participants of that contest even started reading the statement of his problem.

After the contest, Chairman of the program committee announced that he refused to take part in the preparation of contests as long as such inappropriate people as Vasechkin worked on the program committee. That was how Vasechkin became the new Chairman of the program committee, and now he is preparing the next programming contest.

Vasechkin has decided that this time the names of all the problems will consist of the letters «`a`», «`b`», and «`c`» only and the length of each name will be equal to n. In addition, the names must be *extremely complex*. A name is extremely complex if none of its substrings consisting of at least two symbols is a palindrome. Help Vasechkin find all extremely complex names for the problems of the contest.

Input

The only input line contains an integer n ($1 \leqslant n \leqslant 20\,000$).

Output

Output all different extremely complex names of length n consisting of the letters «`a`», «`b`», and «`c`» only. The names should be given in the alphabetical order, one per line. If the total length of the names exceeds 100 000 letters, output the only line «`TOO LONG`».

Example

input.txt	output.txt
2	ab ac ba bc ca cb

Problem G. Computer Security

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

There was an emergency at *ZZZ Inc.* — its secret developments became known to a competing company! Of course, the primary suspects were the employees of *ZZZ Inc.*. System administrator Zhuchkov was asked to report who of the employees had downloaded secret information from the server during the week preceding the incident. Zhuchkov looked through the logs and found the personal number and the data access code of the person who had done that. The administrator gave this information to the company's management, and the same evening the guilty employee disappeared without a trace and Zhuchkov was given a bonus.

However, it soon turned out that Zhuchkov had mixed up some symbols both in the personal number and in the data access code and the sacked employee had in fact been innocent. To avoid such annoying mistakes in the future, Zhuchkov had decided to find all the pairs of employees with similar numbers and check how much their data access codes were similar.

Zhuckov considers two personal numbers *similar* if one of them can be obtained from the other by inserting, deleting, or replacing one digit. Personal numbers do not contain leading zeros. The employees of *ZZZ Inc.* are numbered by consecutive integers starting from 1, and their data access codes are lines consisting of four hexadecimal digits. For each pair of employees with similar personal numbers, Zhuchkov wants to calculate the number of positions in which their data access codes differ. Help Zhuchkov.

Input

The first line contains the number n of employees in *ZZZ Inc.* ($2 \leqslant n \leqslant 65\,536$). The i-th of the following n lines contains the data access code of the employee whose personal number is i. The data access codes consist of digits and lowercase English letters; they are different for different employees.

Output

Output four space-separated integers. The i-th integer should be the number of pairs of employees whose personal numbers are similar and whose data access codes differ in i positions.

Example

input.txt	output.txt
3 dead beef f00d	0 0 2 1

Problem H. Faryuks

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

You want to make your Christmas table special and that's why you've gotten into a plantation of faryuks, which is the only one in the world. You want to take home some of this mysterious fruit. However, it's not that easy to steal faryuks. The plantation is surrounded by a high fence, and the only way to get in or out is through the checkpoint. You can bring as many faryuks as you want to the plantation, but it is strictly forbidden to take them out. This fruit has a very strong aroma, and the automatic aroma sensor at the checkpoint will raise an alarm if the aroma of faryuks is detected. Fortunately, you know that there is an error in the aroma-detecting software. Because of the use of one-byte variables, it does not raise an alarm it the total aroma strength of all the faryuks being taken out is an integer divisible by 256.

You have a bottle of scent and a can of deodorant with you. When a faryuk is sprayed upon with the deodorant, its aroma strength decreases by 25 percent. When a faryuk is scented, its aroma strength increases by 10 percent. (In both cases, the aroma strength can become fractional). The amount of each of the substances you have is enough for treating only one faryuk. A faryuk can be treated first with one substance and then with the other. The effects of the deodorant and of the scent are short. After you take a faryuk through the checkpoint, its aroma strength restores to the initial value.

You've come to the plantation empty-handed and you want to take out the n faryuks you like most of all. Will you be able to do that?

Input

The first line contains the number n of faryuks you want to take out ($1 \leqslant n \leqslant 12$). The second line contains n integers in the range from 1 to 1 000, which are their aroma strengths.

Output

If it is impossible to take out from the plantation all the faryuks that you want, output «`IMPOSSIBLE`».

Otherwise, output in the first line the number of actions necessary for carrying out your plan. Then output the sequence of actions, one action per line. If you want to take out from the plantation $k > 0$ faryuks with the numbers $a_1, \ldots, a_k$, output the line «`take` k a_1 a_2 ... a_k». Similarly, if you want to return to the plantation $s > 0$ faryuks with the numbers $b_1, \ldots, b_s$, output the line «`return` s b_1 b_2 ... b_s». If you want to spray upon with the deodorant the faryuk with the number p, output the line «`dearomatize` p». If you want to scent the faryuk with the number q, output the line «`aromatize` q». The faryuks are numbered from 1 to n in the order in which they are given in the input.

The number of actions in the sequence can't exceed 20 000. If there are several ways to take out all the faryuks, output any of them such that the number of exiting the checkpoint with faryuks is minimal.

Examples

input.txt	output.txt
3 36 56 200	4 take 2 2 3 return 1 3 aromatize 3 take 2 1 3
1 100	IMPOSSIBLE

Problem I. Deer is Better!

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

A crow is sitting on a branch with a piece of cheese in its mouth.
A fox is running by:
"Crow, are you going to the election?"
"No!"
The cheese falls down. The crow sits thinking:
"Would anything have changed had I said yes*?*

A Chukcha set out to go to the election. There are l kilometers between his yaranga and the polling station. His deer run so that they cover every k-kilometer segment of the way in exactly h hours. What is the minimal and maximal time in which the Chukcha will be able to perform his civic duty?

Input

The only input line contains space-separated integers l, k, and h ($1 \leqslant k \leqslant l \leqslant 1\,000$; $1 \leqslant h \leqslant 1\,000$).

Output

Output the minimal and maximal time in which Chukcha can get from his yaranga to the polling station, accurate to at least 10^{-5}. The numbers must be separated with a space.

Example

input.txt	output.txt
30 10 1	3.00000000 3.00000000

Problem J. Communication Fiend

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

Kolya has returned from a summer camp and now he's a real *communication fiend*. He spends all his free time on the Web chatting with his friends via ICQ. However, lately the protocol of this service was changed once again, and Kolya's client stopped working. Now, in order to communicate with his friends again, Kolya has to upgrade his client from version 1 to version n.

Kolya has found m upgrade programs on the Web. The i-th program upgrades the client from version x_i to version y_i and its size is d_i megabytes. Each program can be installed on the corresponding version of the client only; it can't be installed on either earlier or later versions.

The first version, which is currently installed on Kolya's computer, is licensed, and many of the upgrade programs are pirate copies. If a pirate upgrade program is used, the client will always be pirated after that, whatever upgrade is used later. Some of the licensed upgrade programs can be installed on a pirate version of the client, and some of them can't. All the pirate upgrade programs can be installed on both licensed and pirate versions of the client.

Kolya is missing his friends very much, so he wants to download the necessary upgrade programs as soon as possible. Unfortunately, his Web connection is not very fast. Help Kolya determine the minimal total traffic volume necessary for upgrading the client from version 1 to version n. Kolya doesn't care if the final version n of his client is licensed or not.

Input

The first line contains space-separated integers n and m ($2 \leqslant n \leqslant 10^4$; $1 \leqslant m \leqslant 10^4$).

Each of the following m lines describes one upgrade program in the form $x_i\ y_i\ d_i\ s_i$. Here, s_i is the type of the program: «`Pirated`», «`Cracked`», or «`Licensed`». A cracked upgrade program is a licensed program that can be installed on a pirate version of the client, and a licensed program can't be installed on a pirate version. The numbers x_i and y_i mean that the program is installed on version x_i of the client and upgrades it to version y_i. The number d_i is the size of the program in megabytes ($1 \leqslant x_i < y_i \leqslant n$; $1 \leqslant d_i \leqslant 10^6$). The data in each line are separated with exactly one space.

Output

If Kolya can upgrade the client from version 1 to version n, output «`Online`» in the first line and the minimal necessary total incoming traffic volume in the second line.

If it is impossible to upgrade the client, output «`Offline`».

Examples

<table>
<tr><th>input.txt</th><th>output.txt</th></tr>
<tr><td><pre>3 4
1 3 10 Licensed
1 2 2 Pirated
2 3 3 Licensed
2 3 6 Cracked</pre></td><td><pre>Online
8</pre></td></tr>
<tr><td><pre>3 1
1 2 10 Licensed</pre></td><td><pre>Offline</pre></td></tr>
</table>

Problem K. Imagination (Division 1 only!)

Input file:	`input.txt`
Output file:	`output.txt`
Time limit:	2 seconds
Memory limit:	64 megabytes

Has run out.

Input

The first line contains the number of test cases d ($1 \leqslant d \leqslant 50$).

Each of the following d lines contains space-separated integers p, a, and m ($2 \leqslant p < 10^9$; $0 \leqslant a < p$; $1 \leqslant m \leqslant 20$; $m < p$). The number p is prime.

Output

Output one line for each test case. If there exists a positive integer $n < 10^{18}$ such that $n^n + n^m$ modulo p is equal to a, output the word «`YES`», a space, and any of the numbers n satisfying these constraints. Otherwise, output «`NO`».

Example

input.txt	output.txt
2 11 3 1 11 8 2	YES 567 YES 2