Problem A. Lesson

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

The teacher wrote an integer N on the board. Then Alice asked him to square this integer, after that Bob asked to decrement the result by 1, then Clara asked to multiple the result by N again.

Your task is to find the remainder from division of the resulting integer by 6.

Input

One integer N ($-10^9 \le N \le 10^9$).

Output

Number appeared on the board after the teacher performed operations from Alice, Bob and Clara, modulo 6.

Example

standard input	standard output
2	0

Problem B. Eligibility Tree

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Player is eligible for the ICPC, if he meets Basic Requirements and the Eligibility Requirements.

Basic requirements are:

- A student who has competed in two World Finals is NOT eligible to compete.

- A student who has competed in five Regional Contests is NOT eligible to compete.

Eligibility requirements are:

- A student who meets the Basic Requirements and FIRST began post-secondary studies in 2013 or later is eligible to compete.

- A student who meets the Basic Requirements and was born in 1994 or later is eligible to compete.

You are given four integers: number of contestant's participation in the World Finals F, number of contestant's participation in the Regional Contests R, year of his post-secondary studies (if he started in 2017, year is equal to 1, in 2016 — to 2 and so on) and contestants' year of birth.

Input

The input consists of four integers, each integer on the new line: number of World Finals for the contestant F ($0 \le F \le 2$), number of Regional Contests for the contestant ($F \le R \le 5$), year of contestant's post-secondary studies D ($R \le D \le 7$) and year of contestant's birth Y ($1990 \le U \le 2000$).

Output

Print 1, if the contestant is eligible for ICPC, and 0 otherwise.

Example

standard input	standard output
1 4 4 1990	1
1 4 6 1990	0
0 4 6 1994	1
2 2 3 2000	0
0 5 1 2017	0

Problem C. Shop

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Online book shop started the wholesale. If someone buys two books with prices $p_a \geq p_b$, then he pays only p_a, cheaper book goes to him as a gift.

Zenyk selected for Marichka N books and wants to know minimal sum he must to pay in this shop for these books.

Input

First line of the input consists of one integer N — number of books Zenyk wants to buy ($1 \leq N \leq 10^4$). Next line contains N integers p_i — prices of the books ($1 \leq p_i \leq 10^4$).

Output

Print one ingteger — minimal amount of money Zenyk must pay for selected books.

Example

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

Problem D. Palindromic String

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

String s is lexicographically less, than string r if and only if:

- s is prefix of r or

- leftmost character where strings differ (comparing from left to right) for string s is less than for string r.

Consider all strings consisting of lowercase English letters of length no more than N. Find N-th lexicographically minimal palindromic string in this set.

Input

Input consists of one integer N ($1 \leq N \leq 10^4$).

Output

Print N-th lexicographically minimal palindromic string.

Example

standard input	standard output
1	a

Problem E. Mixed-Radix Numbers

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

In the *mixed-radix numeric systems* base varies from position to position. Each base is the integer not less than 2. So for each position defined the base b_i and the digits at this position may be in the range between 0 and $b_i - 1$. For example, if for lowest position base is equal to 5, for second — to 6 and for third — to 7 (we are counting positions from less significant one), then 42 in this system is represented as 122 (because $42 = 2 + 2 \cdot 5 + 1 \cdot 5 \cdot 6$); but if we want represent 2017 in this system, we will fail because we do not know base for the fourth position.

Given an integer x in decimal notation and set of the bases, convert the integer in the mixed-radix system or tell that there is an error.

Input

First line of the input contains one integer x ($1 \le x \le 10^9$) — number to be converted in the irregular numeric system. Second line contains one integer k ($1 \le k \le 32$) — number of given bases. Third line contains bases — k integers b_i ($2 \le b_i \le 62$). b_i define the radix for i-th position (i.e. i-th from the right digit).

Output

If there is not enough bases to convert given integer to the mixed-radix system, print "**Error**". Otherwise print the result of conversion in next format: for digits between 0 and 9 plain decimal digitals must be used, for digits between 10 and 35 — lowercase English letters from 'a' to 'z' in the increasing order, for digats between 36 and 61 — uppercase English letters between 'A' and 'Z' (for example, "**1Ya**" denotes the integer with 1 in most valuable position, 60 in the second position and 10 in the lowest position).

Examples

standard input	standard output
42 3 5 6 7	122
2017 3 5 6 7	Error
1962 3 16 62 2	1Ya
79477827 6 31 32 33 60 42 2	Error

Problem F. Parsing the Text

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Given a text, count number of words which can be found W or more times and number of sentences which can be found S or more times.

Formal definitions:

- *Text* is non-empty sequence of sentences, separated by one space.

- *Sentence* is non-empty sequence of words, separated by one space or by comma ('**,**') and one space. Sentence is terminated either by '!', '?' and '.'.

- *Word* is non-empty sequence of lower- or uppercase English letters. Text is case-sensitive, i.e. same lower- and uppercase letters considered to be distinct.

Note that if two sentences differ only by the final character, they considered to be the same, but if they differ by atleast one comma inside, they considered to be different.

Input

First line of the input contains two integers W and S ($1 \le W, S \le 10^9$). Second line contains the text, length of the text does not exceed 10^3.

Output

Print two integers — number of distinct words which can be found in the text atleast W times and number of distinct sentences which can be found in the text atleast S times.

Example

standard input
2 3
Kto tam? It is Pechkin. Kto tam? My name is Fedor. Kto tam? Pechkin!

standard output
4 1

standard input
2 2
Hello, how are you? You! Hello how are you? Hello how ARE you?

standard output
4 0

Problem G. Card Game

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Zenyk and Marichka start playing Marichka's favourite card game. But Zenyk forgets even how the deck looks like. Of course, he can't just ask Marichka.

He knows that deck contains N cards. Also there are some number of suits, let's say it's K. Number of cards of each suit is the same. Suits are numbered from 1 to K.

At the beginning of the game M cards were dealt. So Zenyk knows suits of these cards. Help Zenyk to find if it's possible to determine the value of K uniquely. Note that the deck is valid so at least one valid value of K exists.

Input

The first line contains 2 integers – N and M ($1 \le N \le 10^9$, $1 \le M \le \min(N, 10^5)$).

The second line contains M integers A_i – suit of the i-th card ($1 \le A_i \le N$).

Output

Print "YES" if it's possible to determine number of suits uniquely and "NO" otherwise.

Examples

standard input	standard output
36 11 1 4 2 4 4 2 4 1 4 4 4	YES
4 2 1 1	NO

Note

In the first test the only valid situation is 4 suits 9 cards each.

In the second test there can be 1 or 2 suits.

Problem H. The Most Expensive Gift

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Zenyk wants to make the best gift in the world for Marichka. He is visiting a gift shop and wants to buy the most expensive one.

But this is not a typical store, they have only string S consisting of letters 'a', 'b' and 'c'. Zenyk can choose any subsequence of this string as a gift. Subsequence T is a string that can be derived from S by deleting some (possible none) letters without changing the order of the remaining letters. Price of subsequence T equals $\frac{len_T^2}{c_T}$, where len_T is the length of subsequence T and c_T is the length of the smallest cycle of T.

String R is the cycle of string T if

- $length(R)$ is a divisor of $length(T)$.

- $R_{i\%length(R)} = T_i$ for all $i \in [0, length(T) - 1]$ (indexing from 0).

Help Zenyk to find the most expensive gift.

Input

The first line of the input contains one integer N, which denotes the length of the string S ($1 \leq N \leq 10^4$). Second line contains string S consisting of letters 'a', 'b' and 'c'.

Output

Print one integer – the price of the most expensive gift.

Example

standard input	standard output
11 abcabacbcac	18

Note

One of the most expensive subsequences is "ababab". It's length equals 6 and length of the smallest cycle equals 2. So price of this subsequence equals $\frac{6^2}{2} = 18$. There are another subsequences with the same price.

Problem I. Cut the Cake

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

It is Marichka's k^2-th birthday today! Zenyk bought a big cake for this occasion and now he wants to cut it.

For the sake of simplicity, consider the cake as rectangular matrix with n rows and m columns. There are exactly k^2 candles on it, each of them located in a unique cell of the matrix. Zenyk wants to cut the cake with $k - 1$ horizontal and $k - 1$ vertical cuts. (Note that he's only allowed to cut between cells.) After the cutting, each of k^2 parts must contain a single candle.

You task is to find and output any valid cutting, or indicate that it's impossible to achive the goal.

Input

The first line contains three integers n, m and k ($2 \le k \le n, m \le 200$). The following n lines contain a string of m characters each. Character '1' represents a cell with a candle on it, while '0' respresents a cell without candle.

It's guaranteed that there are exactly k^2 candles on the cake.

Output

In the first line print "YES" if it's possible to cut the cake the way Zenyk wants, otherwise print "NO".

In case of positive answer the second line must contain $k - 1$ unique valid indices of the horizontal cuts, and the third line must contain $k - 1$ unique valid indices of the vertical cuts. A cut between rows (or columns) i and $i + 1$ has index i (1-based).

Examples

standard input	standard output
4 4 2 1000 0001 0010 0001	YES 2 3
3 4 2 1110 0000 0100	NO

Problem J. Bad Word

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Marichka has seen bad word while surfing the net! She immediately starts crying and asks Zenyk to destroy it.

Zenyk knows that Marichka saw word S which consists of lower case english characters. Zenyk can delete any substring $S_l S_{l+1} S_{l+2} \ldots S_r$ of this word in one minute. But he knows that Marichka is keen on palindromes so if this substring is palindrome, Marichka will resent. Zenyk decided that he wouldn't delete such substrings. Now Zenyk wants to know minimum time to destroy bad word or if it is impossible.

Palindrome is such string that reads the same backward as forward. For example, strings "bob", "abba", "aaaa" are palindromes and "cat", "dog", "penguin" are not.

Input

First line contains one integer N – length of the word S ($1 \le N \le 10^5$). Second line contains word S which consists of lower case english characters.

Output

Print minimum number of minutes to destroy bad word or -1 if it is impossible.

Examples

standard input	standard output
7 abcdcba	2
3 xxx	-1

Note

In the first case Zenyk can delete substring "bcd" during the first minute. Remaining word equals "acba" and can be deleted during the second minute.

Problem K. Frog Jumping

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	256 mebibytes

Marichka and Zenyk like romantic evenings. Today they decided to watch frogs jump over stones.

There are n stones placed in a line and numbered from the left to the right using integers from 1 to n, inclusive. The distance between any two consecutive stones is exactly 1 meter.

There are also m frogs, initially located on the first stone. The objective is to move all frogs to the last (n-th) stone by jumping. Each frog can only jump forward.

The following two condition must be fulfilled:

1. Stones a_1, a_2, ..., a_k must be visited by exactly one of the frogs.

2. All the other stones (except the first one and the last one) must be never visited by any frog.

When the i-th frog jumps more than d meters in a single jump, it costs c_i units of energy. Any smaller jump costs nothing.

Your task is to find the minimum total amount of energy needed for all frogs to get to the last stone.

Input

The first line of the input contains four space-separated integers n, m, k and d ($3 \le n \le 10^9$, $1 \le m, k \le 10^5$, $1 \le d \le 10^9$). The second line contains m space-separated integers c_i, which are the energy costs of a big jump for the corresponding frogs ($1 \le c_i \le 10^9$). The third line contains k space-seperated unique integers a_i, which are the indices of stones that must be visited exactly once ($2 \le a_i < n$).

Output

In the first and only line of the output print a single integer — minimum total energy cost.

Examples

standard input	standard output
10 2 3 3 4 7 4 8 7	4
10 2 2 3 4 7 9 5	15

Problem L. Impress Her

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1.5 seconds
Memory limit:	256 mebibytes

Zenyk decided to impress Marichka and solve the following interesting problem.

You are given a matrix of size $n \times m$ filled with integers. It's guaranteed that all cells which contain the same value are 4-side connected.

Let's define a convex hall of a connected component as minimum-area rectangle (with sides parallel to the matrix sides) that covers all cells of the component. The task is to count the number of pairs of components a and b for which the convex hall of a is inside the convex hall of b. Please note that the convex halls may touch by the side.

Input

The first line contains a pair of integers n and m ($1 \le n, m \le 500$) — the number of rows and columns of the matrix. The next n lines contain m integers each, which represent the matrix. It's guaranteed that matrix integers will be non-negative and won't exceed 10^6.

Output

In the only line print a single integer — the answer to the problem.

Example

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