Problem A. Poluprimes

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

One day, in a land far far away, Domagoj was brushing up on his number theory knowledge...

And while reading he invented poluprimes!

Poluprimes are numbers that have exactly four divisors. Now to give him trouble, Lovro has given him two numbers l and r, and asked him to count the number of poluprimes x such that $l \leq x \leq r$. Domagoj was never good with computational tasks, so he has asked you to help him!

Input

The first line of input contains two integers, l and r, $(1 \leq l \leq r \leq 5000)$, the numbers given by Lovro.

Output

In a single line, output the number of poluprimes between l and r.

Examples

standard input	standard output
2 11	3
13 20	2

Problem B. Permutation

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

For his birthday, Petar was given a permutation!

A permutation is a sequence of integers of length n, such that all integers from 1 to n appear exactly once. Petar can in one move swap any two adjacent numbers. Now Petar is interested in whether he can sort his permutation in an even number of moves (make an even number of swaps of adjacent numbers such that he achieves the permutation $1, \ldots, n$). Petar is still little, so he asked you for help!

Input

The first line contains the integer n ($1 \leq n \leq 100$) - the length of the permutation.

The second line contains n integers $p_1, \ldots, p_n$ representing the permutation.

Output

In a single line output either "Yes"or "No"(without quotes).

Examples

standard input	standard output
2 2 1	No
3 2 3 1	Yes

Problem C. Special Lands

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

Dominik has found himself in the towns of Istria. The road map of Istria can be modeled as a graph where the nodes represent the towns and the edges represent the roads. There are n towns in Istria connected by m roads and at most one road between any pair of cities. All roads are bidirectional. Dominik likes unique things, so he is interested in "special graphs". To be more precise, he is interested in whether for every pair of different nodes, there exists a unique path between them. A path is a sequence of distinct edges which connects two nodes. As the road map of Istria can be very large, he has asked you to help him decide if Istria is special.

Input

The first line of input contains two integers n and m ($1 \leq n, m \leq 1000$). The next m line contains two different integers u and v ($1 \leq u, v \leq n, u \neq v$), description of the roads in Istria.

Output

In a single line output either "Yes"or "No"(without quotes).

Examples

standard input	standard output
3 2 1 2 2 3	Yes
4 4 1 2 2 3 3 4 2 4	No

Problem D. Grid

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

Luka is stuck in a grid! The grid can be imagined as a $m \times n$ table, where some cells are empty and some are blocked. In the beginning, Luka can choose an empty cell to begin his journey, and then in each move, he can choose one of the four adjacent cells and move to it if it is empty and he hasn't visited it before. Luka wants his journey to last as long as possible, so please calculate the largest number of cells Luka can visit!

Input

The first line contains integers r and c ($1 \leq r, c \leq 5$), denoting the number of rows and columns of the grid.

The following n lines contain descriptions of the rows of the grid, empty cells are marked with `'.'` and blocked cells with `'#'`.

Output

In a single line output the largest number of cells Luka can visit.

Examples

standard input	standard output
3 3 ... #.# ...	5
4 4 ###. #... #.##	8

Problem E. Roads

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	512 megabytes

This time Dominik is in Northern Croatia! We can again imagine Northern Croatia as an undirected graph, and this time it is additionally weighted. In other words, each road has a certain length. There are n cities on the map connected by m roads. Dominik is currently in city s and has to arrive in the city t as soon as possible to make it to a wedding... However, his superstitious friend Paula demands that the number of roads Dominik passes through be divisible by 3. Please help Dominik determine the length of the shortest possible route with which Paula will be happy as well. Dominik can decide to pass through the same road multiple times.

Input

The first line contains integers n, m, s and t ($1 \leq n \leq 100\,000, 1 \leq m \leq 300\,000, 1 \leq s, t \leq n$). The following m lines contain descriptions of the roads a_i, b_i, c_i ($1 \leq a_i, b_i \leq n, 1 \leq c_i \leq 10^9$) - it means a road of length c_i connects cities a_i and b_i.

Output

In the only line, output the length of the shortest possible route under Paula's conditions. If there is no such route, output " -1 " without quotes.

Examples

standard input	standard output
3 3 1 3 1 2 1 2 3 1 3 1 2	4
6 5 1 5 1 2 3 2 3 6 1 5 7 3 5 2 4 5 1	9
2 0 1 2	-1

Problem F. Data structure

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

This is a classical data structure. You are given an array $a_1, \ldots, a_n$ of size n. You have to support the following three operations.

`1 l r` - for each $l \leq i \leq r$ set $a_i := \varphi(a_i)$ where φ is the Euler totient function.

`2 l r` - output the sum of a_i such that $l \leq i \leq r$

`3 l r` - output the maximum value of a_i such that $l \leq i \leq r$

Input

The first line contains two integers n and q ($1 \leq n, q \leq 200\,000$) where n is the length of the array and q is the number of operations.

The next q lines contain descriptions of the operations as described in the problem statement.

Output

For each operation of type 2 or 3 output the answer.

Examples

standard input	standard output
3 3 7 5 11 1 2 3 2 2 3 3 1 3	14 10
5 5 6 3 11 7 2 2 2 5 3 1 5 1 1 2 2 1 3 3 1 4	23 11 15 11

Note

In the first case, the array at the beginning looks like this:

a : 7 5 1

And, because $\varphi(5) = 4$ and $\varphi(11) = 10$, after the first operation it looks like:

a : 7 4 10

Now the sum of the second and third elements is $4 + 10 = 11$.

And the maximum of all of the elements is 10.

Problem G. Palindromes

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

Mila really likes palindromes. She was given a string s. She is interested in the number of triples of numbers (a, b, c) such that $(1 \leq a \leq b < c \leq |s|)$ and the substring from the a-th to the b-th position is a palindrome as well as the substring from the $b+1$-th position to the c-th position. In other words, she is interested in the number of neighboring pairs of palindromes.

Input

The first line contains the string s $(1 \leq |s| \leq 10^6)$.

Output

In a single line, output the number of such triples.

Examples

standard input	standard output
abbaba	12
xyz	2

Note

In the first example, the possible triples (a, b, c) are:

$(1, 1, 2), (1, 1, 3), (1, 4, 5)$ $(2, 2, 3), (2, 2, 4), (2, 3, 4), (2, 3, 6)$, $(3, 3, 4), (3, 3, 6), (3, 5, 6)$, $(4, 4, 5)$, $(5, 5, 6)$

In the second example, the possible triples (a, v, c) are:

$(1, 1, 2), (2, 2, 3)$

Problem H. 4-Cycles

Input file:	`standard input`
Output file:	`standard output`
Time limit:	5 seconds
Memory limit:	1024 megabytes

You are given an undirected simple graph G consisting of n nodes and m edges. Output the number of unordered quadruples of edges such that they form a cycle of length 4.

Input

The first line contains numbers n and m ($1 \leq n, m \leq 50\,000$).

The next m lines contain pairs of numbers u_i and v_i ($1 \leq u_i, v_i \leq n$) describing an edge between nodes u_i and v_i.

Output

In the only line, output the number of such cycles.

Examples

standard input	standard output
4 5 1 2 2 3 3 4 4 1 2 4	1
6 15 1 2 1 3 1 4 1 5 1 6 2 3 2 4 2 5 2 6 3 4 3 5 3 6 4 5 4 6 5 6	45