Problem A. Required Length

Time Limit 2000 ms

Problem Statement

Takahashi wants to set his password for a certain website to a string P consisting of lowercase English letters.
The password for that website must be a string of length at least L.

Determine whether P satisfies the length condition, that is, whether it is a string of length at least L.

Constraints

- P is a string consisting of lowercase English letters with length between 1 and 100, inclusive.
- $1 \leq L \leq 100$
- L is an integer.

Input

The input is given from Standard Input in the following format:

P
L

Output

If P satisfies the length condition, print `Yes`; otherwise, print `No`.

Sample 1

Input	Output
chokudai 5	Yes

The length of `chokudai` is 8, which is at least 5, so it satisfies the length condition.
Thus, print `Yes`.

Sample 2

Input	Output
ac 3	No

The length of `ac` is 2, which is less than 3, so it does not satisfy the length condition.
Thus, print `No`.

Sample 3

Input	Output
atcoder 7	Yes

The length of `atcoder` is 7, which is at least 7, so it satisfies the length condition.
Thus, print `Yes`.

Problem B. G1

Time Limit 2000 ms

Problem Statement

In AtCoder Kingdom, there are N horse races being held.
Horses aged A_i or younger can participate in the i-th race.
Among the N races, how many races can a K-year-old horse participate in?

Constraints

- All input values are integers.
- $1 \leq N \leq 100$
- $1 \leq A_i \leq 100$
- $1 \leq K \leq 100$

Input

The input is given from Standard Input in the following format:

N
$A_1 \ A_2 \ \ldots \ A_N$
K

Output

Output the answer as an integer.

Sample 1

Input	Output
5 3 1 4 1 5 4	2

- Horses aged 3 or younger can participate in the 1st race.
- Horses aged 1 or younger can participate in the 2nd race.

- Horses aged 4 or younger can participate in the 3rd race.
- Horses aged 1 or younger can participate in the 4th race.
- Horses aged 5 or younger can participate in the 5th race.

Among the 5 races, a 4-year-old horse can participate in the 3rd and 5th races, which is 2 races.

Sample 2

Input	Output
1 1 100	0

There may be no races that a K-year-old horse can participate in.

Sample 3

Input	Output
15 18 89 31 2 15 93 64 78 58 19 79 59 24 50 30 38	8

Problem C. Reverse Proxy

Time Limit 2000 ms

Problem Statement

There are N boxes numbered $1, 2, \ldots, N$. Initially, all boxes are empty.

Q balls will come in order.
Takahashi will put the balls into the boxes according to the sequence $X = (X_1, X_2, \ldots, X_Q)$.
Specifically, he performs the following process for the i-th ball:

- If $X_i \geq 1$: Put this ball into box X_i.
- If $X_i = 0$: Put this ball into the box with the smallest number among those containing the fewest balls.

Find which box each ball was put into.

Constraints

- All input values are integers.
- $1 \leq N \leq 100$
- $1 \leq Q \leq 100$
- $0 \leq X_i \leq N$

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \ Q \\
X_1 \ X_2 \ \ldots \ X_Q
\end{array}
$$

Output

If the i-th ball was put into box B_i, output in the following format:

```
B_1 B_2 ... B_Q
```

Sample 1

Input	Output
`4 5` `2 0 3 0 0`	`2 1 3 4 1`

There are 4 boxes, and 5 balls come.

- Initially, all boxes are empty.
  - The numbers of balls in box $1, 2, 3, 4$ are $0, 0, 0, 0$, respectively.
- Since $X_1 = 2$, put the 1st ball into box 2.
  - The numbers of balls in box $1, 2, 3, 4$ are $0, 1, 0, 0$, respectively.
- Since $X_2 = 0$, put the 2nd ball into box 1, which has the smallest number among those containing the fewest balls.
  - The numbers of balls in box $1, 2, 3, 4$ are $1, 1, 0, 0$, respectively.
- Since $X_3 = 3$, put the 3rd ball into box 3.
  - The numbers of balls in box $1, 2, 3, 4$ are $1, 1, 1, 0$, respectively.
- Since $X_4 = 0$, put the 4th ball into box 4, which has the smallest number among those containing the fewest balls.
  - The numbers of balls in box $1, 2, 3, 4$ are $1, 1, 1, 1$, respectively.
- Since $X_5 = 0$, put the 5th ball into box 1, which has the smallest number among those containing the fewest balls.
  - The numbers of balls in box $1, 2, 3, 4$ are $2, 1, 1, 1$, respectively.

The balls were put into boxes $2, 1, 3, 4, 1$ in order. Thus, output `2 1 3 4 1`.

Sample 2

Input	Output
`3 7` `1 1 0 0 0 0 0`	`1 1 2 3 2 3 1`

Sample 3

Input	Output
`6 20` `4 6 0 3 4 2 6 5 2 3 0 3 2 5 0 3 5 0 2 0`	`4 6 1 3 4 2 6 5 2 3 1 3 2 5 1 3 5 4 2 6`

Problem D. Distance Table

Time Limit 2000 ms

Problem Statement

There are N stations, station 1, station 2, $\ldots$, station N, arranged in this order on a straight line.
Here, for $1 \leq i \leq N-1$, the distance between stations i and $(i+1)$ is D_i.

For each pair of integers (i, j) satisfying $1 \leq i < j \leq N$, find the distance between stations i and j.
For the output format, refer to the Output section.

Constraints

- $2 \leq N \leq 50$
- $1 \leq D_i \leq 100$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

N
$D_1 \ D_2 \ \ldots \ D_{N-1}$

Output

Output $N-1$ lines.
On the i-th line $(1 \leq i \leq N-1)$, output $(N-i)$ integers, separated by spaces.
The j-th integer on the i-th line $(1 \leq j \leq N-i)$ should be the distance between stations i and $(i+j)$.

Sample 1

Input	Output
5 5 10 2 3	5 15 17 20 10 12 15 2 5 3

The distances between stations are as follows:

- The distance between stations 1 and 2 is 5.
- The distance between stations 1 and 3 is $5 + 10 = 15$.
- The distance between stations 1 and 4 is $5 + 10 + 2 = 17$.
- The distance between stations 1 and 5 is $5 + 10 + 2 + 3 = 20$.
- The distance between stations 2 and 3 is 10.
- The distance between stations 2 and 4 is $10 + 2 = 12$.
- The distance between stations 2 and 5 is $10 + 2 + 3 = 15$.
- The distance between stations 3 and 4 is 2.
- The distance between stations 3 and 5 is $2 + 3 = 5$.
- The distance between stations 4 and 5 is 3.

Thus, output as shown above.

Sample 2

Input	Output
2 100	100

Problem E. Black Intervals

Time Limit 3000 ms

Problem Statement

There are N squares arranged in a row from left to right. Initially, all squares are painted white.

Process Q queries in order. The i-th query gives an integer A_i between 1 and N, inclusive, and performs the following operation:

> Flip the color of the A_i-th square from the left. Specifically, if the A_i-th square from the left is painted white, paint it black; if it is painted black, paint it white.
> Then, find the number of intervals of consecutively painted black squares.
>
> Here, an interval of consecutively painted black squares is a pair of integers (l, r) $(1 \leq l \leq r \leq N)$ that satisfy all of the following:
>
> - The l-th, $(l+1)$-th, $\ldots$, r-th squares from the left are all painted black.
> - Either $l = 1$, or the $(l-1)$-th square from the left is painted white.
> - Either $r = N$, or the $(r+1)$-th square from the left is painted white.

Constraints

- $1 \leq N, Q \leq 5 \times 10^5$
- $1 \leq A_i \leq N$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \;\; Q \\
A_1 \;\; A_2 \;\; \ldots \;\; A_Q
\end{array}
$$

Output

Output Q lines. On the i-th line $(1 \leq i \leq Q)$, output the answer to the i-th query.

Sample 1

Input	Output
5 7 2 3 3 5 1 5 2	1 1 1 2 2 1 1

Below, the i-th square from the left is referred to as square i.
After each query, the state is as follows:

- After the 1st query, only square 2 is painted black. There is 1 interval of consecutively painted black squares: $(l, r) = (2, 2)$.
- After the 2nd query, squares $2, 3$ are painted black. There is 1 interval of consecutively painted black squares: $(l, r) = (2, 3)$.
- After the 3rd query, only square 2 is painted black. There is 1 interval of consecutively painted black squares: $(l, r) = (2, 2)$.
- After the 4th query, squares $2, 5$ are painted black. There are 2 intervals of consecutively painted black squares: $(l, r) = (2, 2), (5, 5)$.
- After the 5th query, squares $1, 2, 5$ are painted black. There are 2 intervals of consecutively painted black squares: $(l, r) = (1, 2), (5, 5)$.
- After the 6th query, only squares $1, 2$ are painted black. There is 1 interval of consecutively painted black squares: $(l, r) = (1, 2)$.
- After the 7th query, only square 1 is painted black. There is 1 interval of consecutively painted black squares: $(l, r) = (1, 1)$.

Thus, output $1, 1, 1, 2, 2, 1, 1$ separated by newlines.

Sample 2

Input	Output
1 2 1 1	1 0

After the 2nd query, all squares are painted white, so output 0 on the 2nd line.

Sample 3

Input	Output
3 3 1 3 2	1 2 1

Problem F. Conflict 2

Time Limit 2000 ms

Problem Statement

There is one server and N PCs. The server and each PC each hold one string, and initially all strings are empty.

Q queries are given. Each query is in one of the following formats:

- `1 p`: Replace the string of PC p with the string of the server.
- `2 p s`: Append string s to the end of the string of PC p.
- `3 p`: Replace the string of the server with the string of PC p.

Find the final string of the server after processing all queries in the given order.

Constraints

- N, Q are integers
- $1 \leq N, Q \leq 2 \times 10^5$
- For every query, p is an integer and $1 \leq p \leq N$.
- For every query of type 2, s is a string of length at least 1 consisting of lowercase English letters.
- The sum of the lengths of s over all queries of type 2 is at most 10^6.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \ Q \\
\text{query}_1 \\
\text{query}_2 \\
\vdots \\
\text{query}_Q
\end{array}
$$

Here, query_i represents the i-th query and is given in one of the following formats:

```
1 p
```

```
2 p s
```

```
3 p
```

Output

Output the answer.

Sample 1

Input	Output
2 6 2 1 at 3 1 2 2 on 1 2 2 2 coder 3 2	atcoder

- Initially, the strings of the server and PCs 1, 2 are all empty.
- 1st query: Append `at` to the end of the string of PC 1. At this time, the strings of the server, PC 1, 2 are empty, `at`, empty, respectively.
- 2nd query: Replace the string of the server with the string of PC 1. At this time, the strings of the server, PC 1, 2 are `at`, `at`, empty, respectively.
- 3rd query: Append `on` to the end of the string of PC 2. At this time, the strings of the server, PC 1, 2 are `at`, `at`, `on`, respectively.
- 4th query: Replace the string of PC 2 with the string of the server. At this time, the strings of the server, PC 1, 2 are `at`, `at`, `at`, respectively.
- 5th query: Append `coder` to the end of the string of PC 2. At this time, the strings of the server, PC 1, 2 are `at`, `at`, `atcoder`, respectively.
- 6th query: Replace the string of the server with the string of PC 2. At this time, the strings of the server, PC 1, 2 are `atcoder`, `at`, `atcoder`, respectively.

Thus, the final string of the server is `atcoder`.

Sample 2

Input	Output
100000 3 1 100 2 300 abc 3 200	

The final string of the server is empty.

Sample 3

Input	Output
10 10 2 7 ladxf 2 7 zz 2 7 kfm 3 7 1 5 2 5 irur 3 5 1 6 2 6 ptilun 3 6	ladxfzzkfmirurptilun

Problem G. E [max]

Time Limit 3000 ms

Problem Statement

There are N six-sided dice. The dice are numbered from 1 to N, and the numbers written on the faces of die i are $A_{i,1}, A_{i,2}, \ldots, A_{i,6}$.

Now, all N dice will be rolled simultaneously. Find the expected value, modulo 998244353, of the maximum value among the numbers written on the face that comes up on each die.

For any die, the face that comes up when the die is rolled is chosen independently and uniformly at random.

▶ Finding the expected value modulo 998244353

Constraints

- $1 \le N \le 10^5$
- $1 \le A_{i,j} \le 10^9$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \\
A_{1,1}\ A_{1,2}\ \ldots\ A_{1,6} \\
A_{2,1}\ A_{2,2}\ \ldots\ A_{2,6} \\
\vdots \\
A_{N,1}\ A_{N,2}\ \ldots\ A_{N,6}
\end{array}
$$

Output

Output the answer.

Sample 1

Input	Output
2 1 1 4 4 4 4 1 1 1 3 3 3	332748121

Let x_i be the number written on the face that comes up on die i.

- Case $x_1 = 1, x_2 = 1$: Occurs with probability $\frac{2}{6} \times \frac{3}{6} = \frac{1}{6}$, and the maximum value among the numbers written on the faces that come up is 1.
- Case $x_1 = 1, x_2 = 3$: Occurs with probability $\frac{2}{6} \times \frac{3}{6} = \frac{1}{6}$, and the maximum value among the numbers written on the faces that come up is 3.
- Case $x_1 = 4, x_2 = 1$: Occurs with probability $\frac{4}{6} \times \frac{3}{6} = \frac{1}{3}$, and the maximum value among the numbers written on the faces that come up is 4.
- Case $x_1 = 4, x_2 = 3$: Occurs with probability $\frac{4}{6} \times \frac{3}{6} = \frac{1}{3}$, and the maximum value among the numbers written on the faces that come up is 4.

Thus, the expected value to be found is $\frac{1}{6} \times 1 + \frac{1}{6} \times 3 + \frac{1}{3} \times 4 + \frac{1}{3} \times 4 = \frac{10}{3} \equiv 332748121 \pmod{998244353}$.

Sample 2

Input	Output
2 1 1 1 1 1 1 2 2 2 2 2 2	2

The numbers written on the faces that come up on dice $1, 2$ are always $1, 2$, respectively, and their maximum value is always 2.

Sample 3

Input	Output
8 55 76 80 21 34 28 82 84 2 32 56 17 11 57 37 28 39 18 47 2 97 25 75 29 72 45 22 75 26 81 6 79 16 68 68 40 31 80 68 57 18 55 49 10 63 91 93 40	213725517

Problem H. Contraction

Time Limit 4000 ms

Problem Statement

You are given an undirected graph G_0 with N vertices and M edges. The vertices and edges of G_0 are numbered as vertices $1, 2, \ldots, N$ and edges $1, 2, \ldots, M$, respectively, and edge i $(1 \leq i \leq M)$ connects vertices U_i and V_i.

Takahashi has a graph G and N pieces numbered as pieces $1, 2, \ldots, N$.
Initially, $G = G_0$, and piece i $(1 \leq i \leq N)$ is placed on vertex i of G.

He will now perform Q operations in order. The i-th operation $(1 \leq i \leq Q)$ gives an integer X_i between 1 and M, inclusive, and performs the following operation:

> In G, if pieces U_{X_i} and V_{X_i} are placed on different vertices and there exists an edge e (on G) between them, create a graph G' by contracting that edge. In this case, if self-loops are created, remove them, and if multi-edges exist, replace them with simple edges.
> Then, all pieces that were placed on the two vertices connected by edge e in G are placed on the newly generated vertex by the contraction of e in G'. Pieces that were placed on other vertices in G are placed on the corresponding vertices in G'. Finally, set this resulting G' as the new G.
> If pieces U_{X_i} and V_{X_i} are placed on the same vertex, or if the vertices they are placed on are not connected by an edge, do nothing.

For each of the operations $i = 1, 2, \ldots, Q$, output the number of edges in G after the i-th operation.

▶ Edge Contraction

Constraints

- $2 \leq N \leq 3 \times 10^5$
- $1 \leq M \leq 3 \times 10^5$
- $1 \leq U_i < V_i \leq N$
- $(U_i, V_i) \neq (U_j, V_j)$ if $i \neq j$.
- $1 \leq Q \leq 3 \times 10^5$

- $1 \le X_i \le M$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N\ \ M \\
U_1\ \ V_1 \\
U_2\ \ V_2 \\
\vdots \\
U_M\ \ V_M \\
Q \\
X_1\ \ X_2\ \ \ldots\ \ X_Q
\end{array}
$$

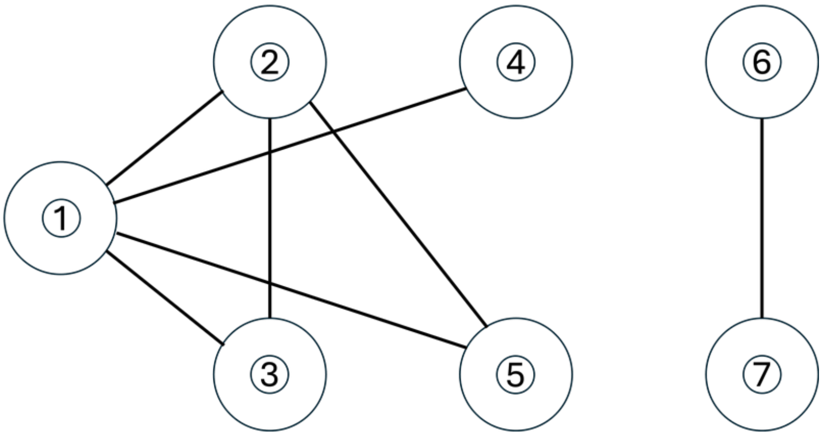
Output

Output Q lines. On the i-th line $(1 \le i \le Q)$, output the number of edges in G after the i-th operation.

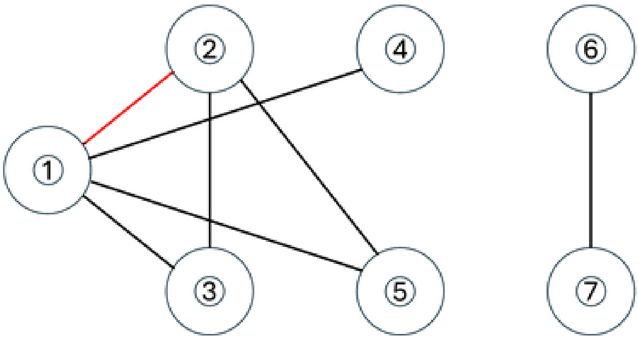
Sample 1

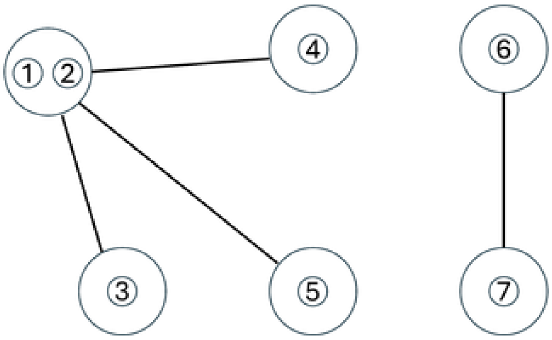
Input	Output
7 7 1 2 1 3 2 3 1 4 1 5 2 5 6 7 5 1 2 3 1 5	4 3 3 3 2

Initially, G is as shown in the figure below. The circled numbers represent pieces with those numbers.



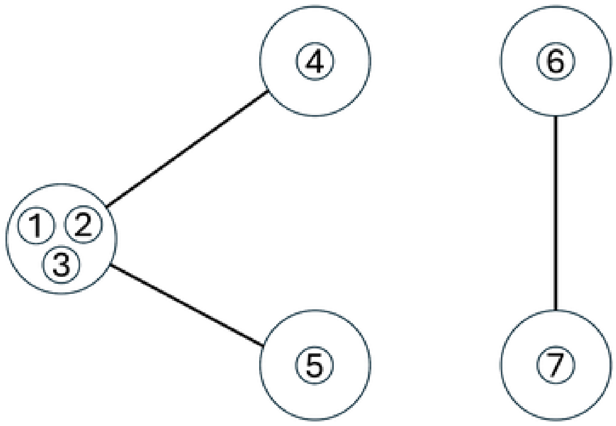
In the 1st operation, we contract the edge between the vertices where pieces 1 and 2 are placed (left figure below).
After the operation, G becomes as shown in the right figure below, and in particular, the number of edges is 4. Note that self-loops have been removed and multi-edges have been replaced with simple edges.

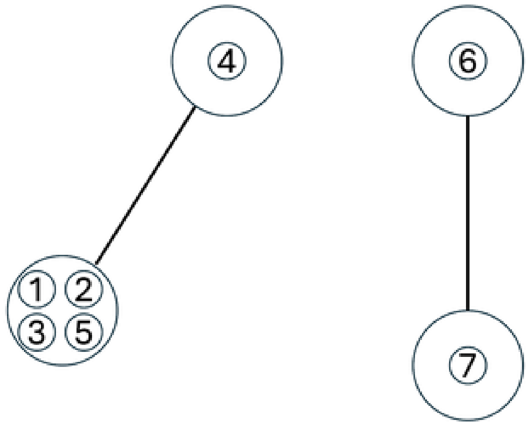




In the 2nd operation, we contract the edge between the vertices where pieces 1 and 3 are placed.
After the operation, G becomes as shown in the left figure below, and the number of edges becomes 3.
In the 3rd operation, since pieces 2 and 3 are placed on the same vertex, G remains unchanged, and the number of edges remains 3.
In the 4th operation as well, since pieces 1 and 2 are placed on the same vertex, G remains unchanged, and the number of edges remains 3.
In the 5th operation, we contract the edge between the vertices where pieces 1 and 5 are placed.
After the operation, G becomes as shown in the right figure below, and the number of edges becomes 2.

Thus, output 4, 3, 3, 3, 2 in this order, separated by newlines.


④
⑥
①②
③
⑤
⑦


④
⑥
①②
③⑤
⑦

Problem I. Count Cycles

Time Limit 6000 ms

Problem Statement

There is an undirected graph G with N vertices and M edges. G does not contain self-loops, but may contain multi-edges. Vertices are numbered from 1 to N, and edges are numbered from 1 to M, with edge i connecting vertices U_i, V_i.

Find the number, modulo 998244353, of cycles contained in G.

More formally, find the number, modulo 998244353, of subsets $\{e_1, e_2, \ldots, e_k\} \subseteq \{1, 2, \ldots, M\}$ $(k \geq 2)$ of the given edge set that satisfy the following condition.

- There exists a permutation $(e'_1, e'_2, \ldots, e'_k)$ of $(e_1, e_2, \ldots, e_k)$ and a vertex sequence $(v_1, v_2, \ldots, v_k)$ such that all of the following hold:
  - $v_1, v_2, \ldots, v_k$ are pairwise distinct;
  - For all j $(1 \leq j \leq k)$, edge e'_j connects vertices $v_j, v_{(j \bmod k)+1}$.

Constraints

- $2 \leq N \leq 20$
- $2 \leq M \leq 2 \times 10^5$
- $1 \leq U_i < V_i \leq N$
- All input values are integers.

Input

The input is given from Standard Input in the following format:

$$
\begin{array}{l}
N \ M \\
U_1 \ V_1 \\
U_2 \ V_2 \\
\vdots \\
U_M \ V_M
\end{array}
$$

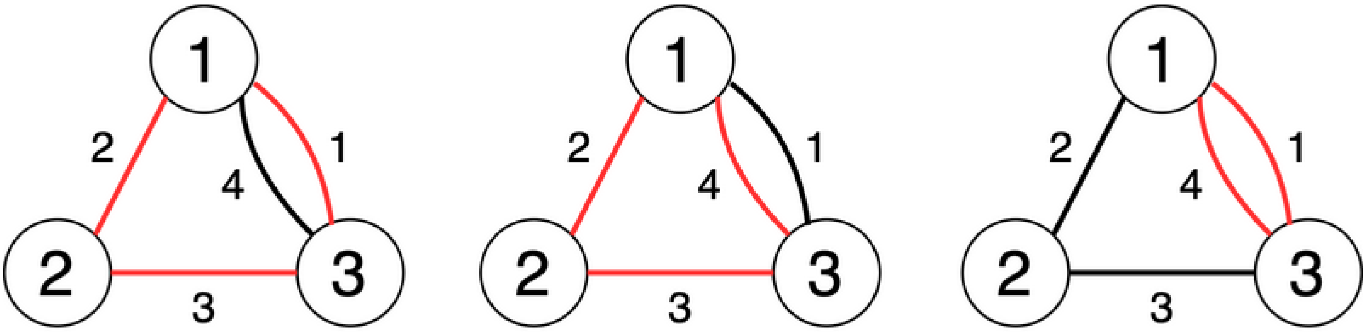
Output

Output the answer.

Sample 1

Input	Output
3 4 1 3 1 2 2 3 1 3	3

As shown in the figure below, there are a total of 3 cycles. The numbers inside the circles and the numbers next to the lines represent vertex numbers and edge numbers, respectively. Red lines represent edges included in cycles, and black lines represent the other edges.



From left to right, these correspond to choosing edge sets $\{1, 2, 3\}$, $\{2, 3, 4\}$, $\{1, 4\}$, respectively.

Sample 2

Input	Output
4 2 1 4 2 3	0

Sample 3

Input	Output
5 15 1 5 3 4 2 3 2 4 3 5 4 5 2 5 2 3 1 3 4 5 2 5 4 5 1 2 3 4 1 5	166