

Power Gems

This question is graded for 0.5%!

Statement

Fluffy the Hamster has recently found n power gems. The gems are numbered from 1 to n .

Fluffy has discovered that he can link several gems together to enhance their power. More precisely, there are m pairs of gems u_i, v_i that he is able to link together. The link between gem u_i and gem v_i has power p_i .

A set of gems that is linked directly or indirectly to one another is called a piece of jewellery. The *quality* of a piece of jewellery is the minimum power of a link that is used to construct the piece of jewellery.

Fluffy wonders what is the maximum quality of a piece of jewellery that contains at least g_i gemstones. If it is impossible to construct such a piece of jewellery, answer -1 .

He has q such queries. Help Fluffy answer them!

Constraints

- $2 \leq n \leq 5 \cdot 10^4$
- $1 \leq m \leq 10^6$
- $1 \leq u_i, v_i \leq n$
- $0 \leq p_i \leq 10^6$
- $1 \leq q \leq 5 \cdot 10^4$
- $2 \leq g_i$

Input

The first line of input will contain three integers n, m, q .

The next m lines of input will each contain three integers, u_i, v_i, p_i . Each triple describes a possible link $u_i \leftrightarrow v_i$, with power p_i .

The next q lines of input will each contain a single integer, g_i . This is a query for the best quality jewellery, with size at least g_i .

Output

For each query, print a single integer, the maximum possible quality of a piece of jewellery containing at least g_i gemstones.

Examples

Sample Input	Expected Output
5 4 5 1 2 1 2 3 3 3 4 2 4 1 4 4 1 4 2 3 4 5 7	4 2 2 -1 -1

Notes

1. A skeleton file has been given to help you. You should not create a new file or rename the file provided. You should develop your program using this skeleton file.
2. You are free to define your own helper methods and classes (or remove existing ones) if it is suitable but you must put all the new classes, if any, in the same skeleton file provided.

Skeleton File

You are given the skeleton file `Gems.java`. You should see the following contents when you open the file:

```
/**
 * Name      :
 * Matric. No :
 */

import java.util.*;

public class Gems {
    private void run() {
        // implement your "main" method here
    }

    public static void main(String args[]) {
        Gems runner = new Gems();
        runner.run();
    }
}
```

```
}  
}
```