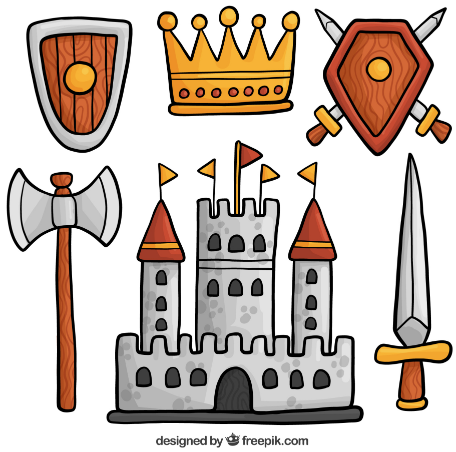
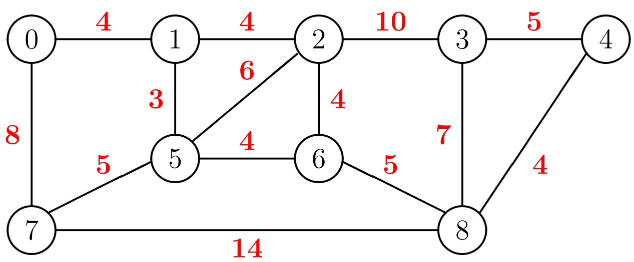
Back to the Middle Ages



Many, many years ago, when people travelled on horseback, journeys often took several days. Luckily, there were inns where they could spend the night. Some travellers did not mind covering long distances each day if it meant reducing the number of days on the road. But others, who were physically more fragile, preferred short daily routes, even if it took them many days to reach their destination.

Places where travellers could spend the night were always located at the endpoints of roads (that is, there were no inns along the roads, which, of course, could be travelled in both directions). Fragile travellers always took n days to make a journey that covered n roads (for $n > 0$). On the first day, they travelled the first road, on the second day, they travelled the second road, and so on, until the last day, when they travelled the final road. For them, the *hardness* of the journey was measured by the longest duration of these daily rides. *Good* journeys were journeys whose hardness was as little as possible.

Let us see some examples in the area depicted below, where there are 9 locations in which travellers could spend the night and 14 roads. The numbers in red are the times (in hours) to travel the roads.



- Journey 2 1 5 would be done in two days: 4 hours of travel on the first day and 3 hours on the second. Its hardness is 4 (because $4 = \max(4, 3)$). Since there are no journeys from location 2 to location 5 in which the longest duration of the daily rides is less than 4, 2 1 5 is a good journey. Notice that journey 2 6 5 is also good (given that $\max(4, 4) = 4$), whereas journey 2 5 is not (because its hardness is 6).

- Journeys 0 1 2 6 8 and 0 1 5 6 8 are both good and their hardness is 5.

Task

Write a program that, given the description of an area, the starting and the destination locations, computes the hardness of the good journeys between them. It is guaranteed that, for the given inputs, there is always some route between any two locations.

Input

The input first line has two integers, L and R, which denote, respectively, the number of locations and the number of roads in the area. Locations are identified by integers, ranging from 0 to $L-1$. Each of the following R lines contains three integers, l_1, l_2 and h, indicating that there is a bidirectional road between the distinct locations l_1 and l_2, which takes h hours to ride.

The next line has a single integer, T, specifying the number of test cases. T lines follow, each one with two different integers, s_i and d_i, which represent the starting and the destination locations in the i^{th} test case (for every $i = 1, \ldots, T$).

Constraints

$2 \leq L \leq 10\,000$	Number of locations
$1 \leq R \leq 50\,000$	Number of roads
$1 \leq T \leq 250$	Number of test cases
$1 \leq h \leq 23$	Time to ride a road (in hours)

Output

The output consists of T lines, each one with a single integer. The number on the i^{th} line represents the hardness of the good journeys from s_i to d_i (for every $i = 1, \ldots, T$).

Sample Input

```
9 14
0 1 4
1 2 4
2 3 10
3 4 5
0 7 8
1 5 3
2 6 4
3 8 7
4 8 4
7 5 5
5 6 4
6 8 5
7 8 14
2 5 6
6
2 5
0 8
1 3
4 7
2 0
5 1
```

Sample Output

```
4
5
5
5
4
3
```