

# SPOJ IITKWPCG

*Help the old King*

**Prof. Edson Alves**

**Faculdade UnB Gama**

Once upon a time there lived a king in a far far country. In the country, there are  $n$  cities and  $m$  roads. He was severely attacked by his enemy. The enemy damaged all the cities of King's country. As the roads between the cities had been damaged, the King wanted to repair those. So he decided to launch a tender for this.

As King's country is a well managed country. By well managed country, we mean that it is possible to go from each city to any other city. But now as the city has been destroyed by enemies, all the roads are broken, the king will like to rebuild the roads in such a way that it is still a well manged country.

Cost of repairing the road in the country is really wierd, it is not addition of costs but it is instead multiplication of those. What it means that if the king decides that he should repair 5 roads, then total cost of repairing those roads will be multiplication of all the 5 costs.

Era uma vez um rei que vivia em um reino muito, muito distante. Neste reino há  $n$  povoados e  $m$  estradas. O rei foi atacado severamente por seu inimigo. O inimigo danificou todos os povoados do reino. Uma vez que as estradas entre os povoados foram danificadas, o rei quer reparar estas estradas. Deste modo, ele decidiu lançar uma proposta para isto.

O reino é bem planejado, e por bem planejado entenda que é possível ir de qualquer povoado a qualquer outro povoado. Mas como os povoados foram destruídos por inimigos, todas as estradas estão em ruínas, o rei irá reconstruir as estradas de modo que o reino se torne novamente um reino bem planejado.

O custo de reparo de uma estrada neste reino é realmente estranho, de modo que ele não adiciona ao custo total, e sim multiplica. Isto significa que se o rei decidir reparar 5 estradas, então o custo total do reparo destas estradas será a multiplicação dos 5 custos individuais de cada estrada.

*As the King's treasure has been damaged by the attack of foreign city, he would like to spend minimum amount of money and that the will want that his country still remains well managed country. Surprisingly the company that was given tender had a rule that all the costs will be in powers of two, as they were really love with binary numbers.*

*As value of the total cost can be really large. We do not want to know the actual cost, instead output number of divisors of the number.*

Uma vez que o tesouro real foi prejudicado pelo ataque estrangeiro, o rei gostaria de gastar o mínimo possível de dinheiro e ele deseja que o país continue sendo bem planejado. Supreendentemente, a companhia que assumiu a proposta tem uma regra que todos os custos serão potências de dois, uma vez que eles realmente gostam de números binários.

Uma vez que o custo total pode ser muito grande, nos não desejamos este valor. Ao invés disso, imprima o número de divisores deste valor.

## Input

*$T$ : number of test cases ( $T \leq 5$ )*

*$n$  and  $m$  ( $n \leq 10^5$  &&  $m \leq 2 \times 10^5$ )*

*Next  $m$  lines will have  $a, b, c$ , which denotes that cities  $a$  and  $b$  are connected with road of cost  $c$ .*

*( $c \geq 2$  &&  $c \leq 10^{18}$  &&  $c$  will always be power of 2) ( $1 \leq a \leq n$ ) and ( $1 \leq b \leq n$ )*

## Output

*For each test case, output a single line containing a number as stated in the problem.*

## Entrada

$T$ : número de casos de teste ( $T \leq 5$ )

$n$  e  $m$  ( $n \leq 10^5$  &&  $m \leq 2 \times 10^5$ )

As  $m$  linhas seguintes terão inteiros  $a, b, c$ , os quais indicam que os povoados  $a$  e  $b$  estão conectadas por uma estrada de custo  $c$ .

( $c \geq 2$  &&  $c \leq 10^{18}$  &&  $c$  será sempre uma potência de 2) ( $1 \leq a \leq n$ ) e ( $1 \leq b \leq n$ )

## Saída

Para cada caso de teste, imprima uma única linha contendo o número solicitado pelo problema.

## **Exemplo de entrada e saída**



## Exemplo de entrada e saída

2 1

## Exemplo de entrada e saída

2 1  
↑  
*# de povoados*

## Exemplo de entrada e saída

2 1  
↑  
*# de estradas*

## Exemplo de entrada e saída

2 1

1

2

## Exemplo de entrada e saída

2 1  
1 2 16  
↑  
*povoado a*

1

2

## Exemplo de entrada e saída

2 1  
1 2 16  
↑  
*povoado b*

1

2

## Exemplo de entrada e saída

2 1  
1 2 16  
↑  
*custo do reparo  $c$*

1

2

## Exemplo de entrada e saída

2 1

1 2 16





## Exemplo de entrada e saída

2 1

1 2 16



## Exemplo de entrada e saída

2 1  
1 2 16  
↓  
5



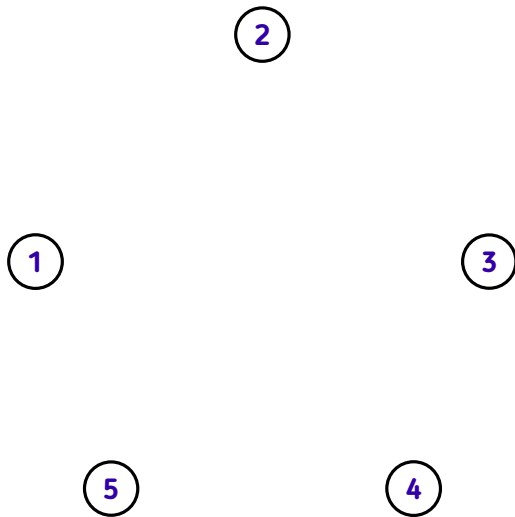
## **Exemplo de entrada e saída**

## Exemplo de entrada e saída

5 5

## Exemplo de entrada e saída

5 5



## Exemplo de entrada e saída

5 5  
1 2 2

1

2

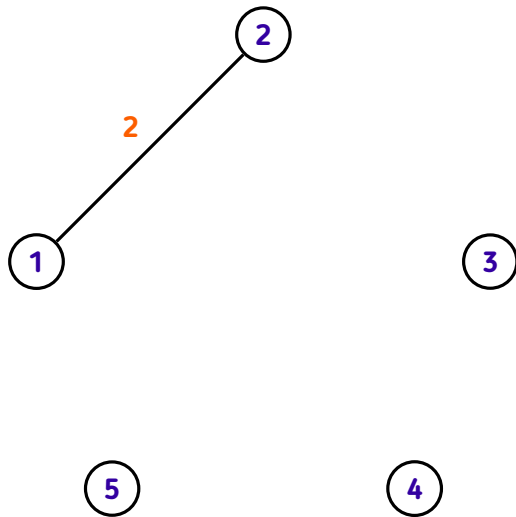
3

5

4

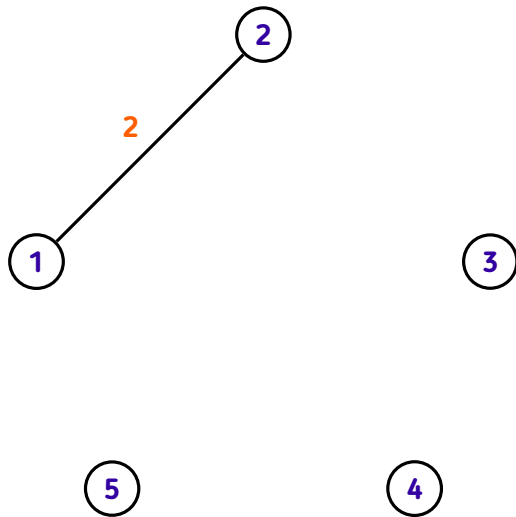
## Exemplo de entrada e saída

5 5  
1 2 2



## Exemplo de entrada e saída

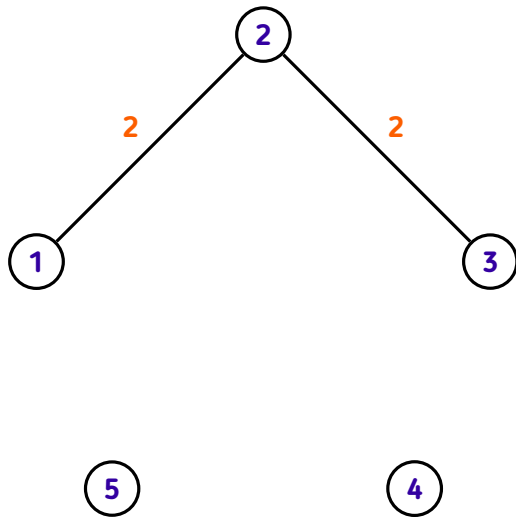
5 5  
1 2 2  
2 3 2





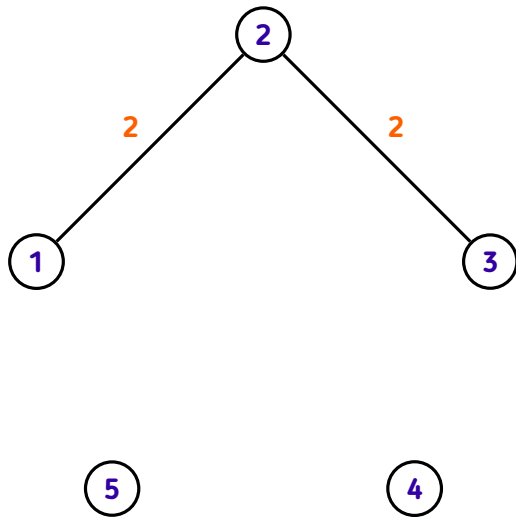
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2



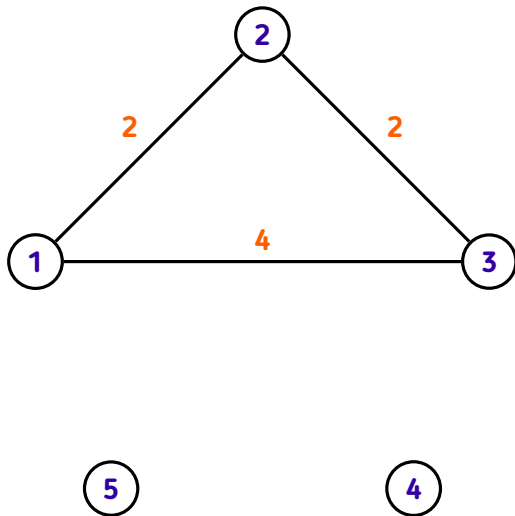
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4



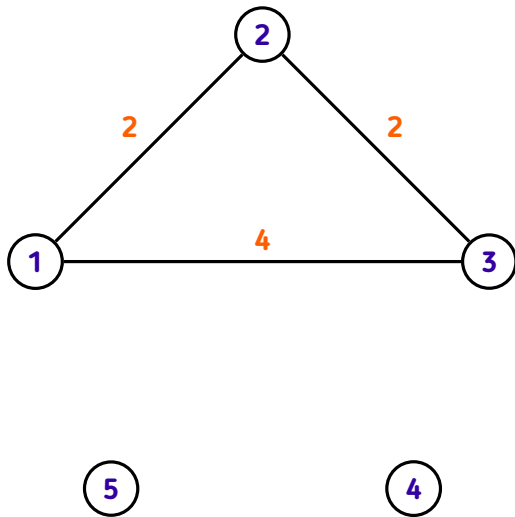
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4



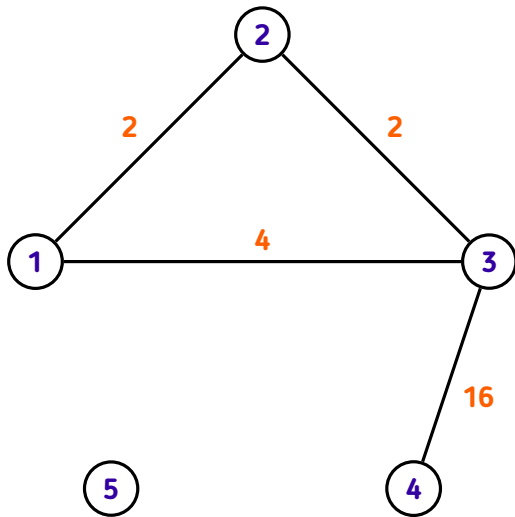
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16



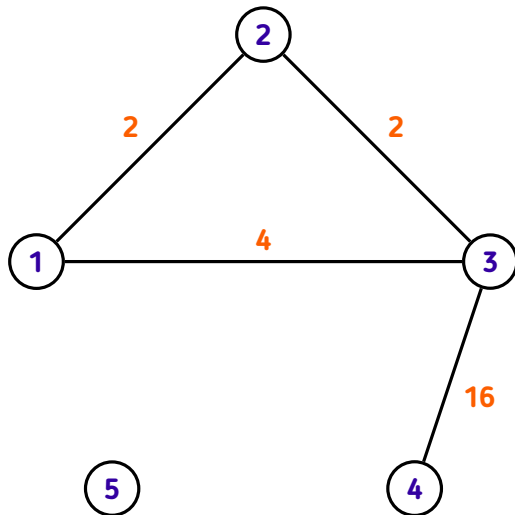
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16



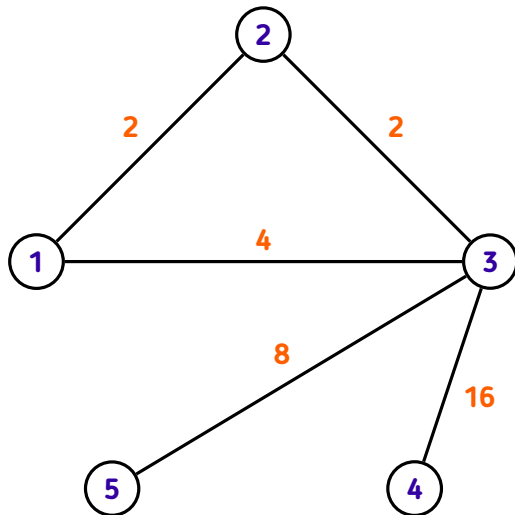
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8



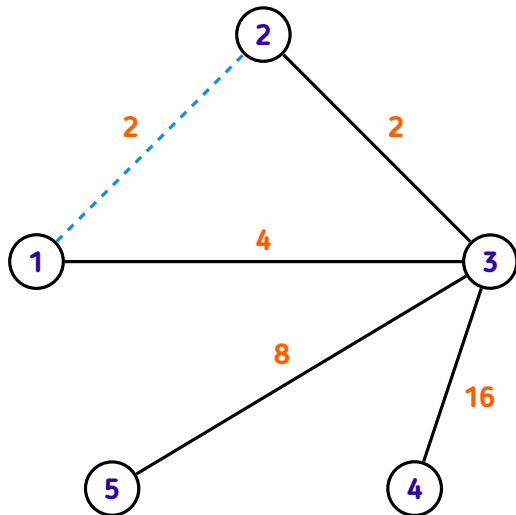
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8



## Exemplo de entrada e saída

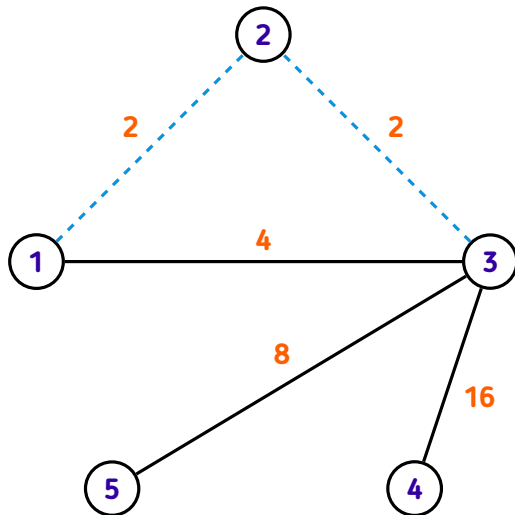
5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8





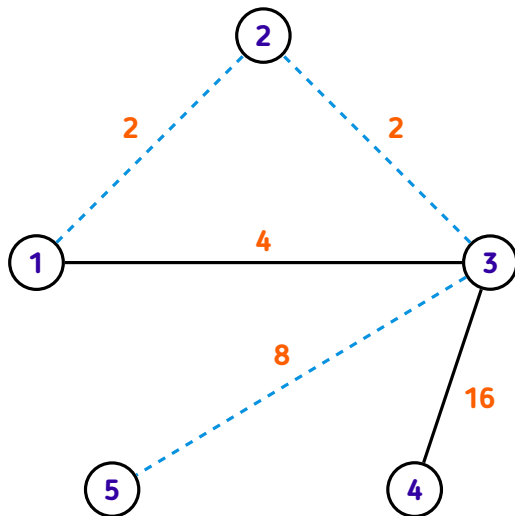
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8



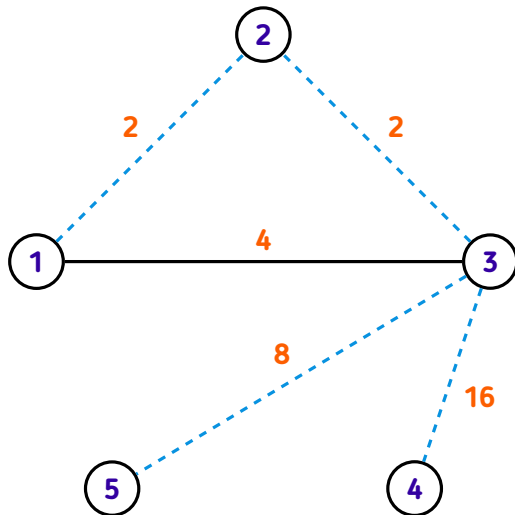
## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8



## Exemplo de entrada e saída

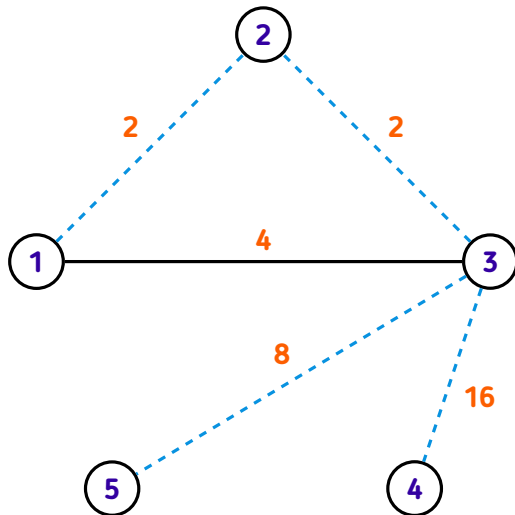
5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8



## Exemplo de entrada e saída

5 5  
1 2 2  
2 3 2  
1 3 4  
3 4 16  
3 5 8

↓  
10



## Solução

# Solução

- ★ A MST minimiza também o produto dos pesos da arestas

## Solução

- ★ A MST minimiza também o produto dos pesos da arestas
- ★ Para verificar este fato, basta substituir os pesos de  $G$  por seus respectivos logaritmos e identificar a MST do grafo  $G'$  resultante

## Solução

- ★ A MST minimiza também o produto dos pesos das arestas
- ★ Para verificar este fato, basta substituir os pesos de  $G$  por seus respectivos logaritmos e identificar a MST do grafo  $G'$  resultante
- ★ Neste problema em particular, por conta das restrições dos pesos, use  $\log_2()$



## Solução

- ★ A MST minimiza também o produto dos pesos das arestas
- ★ Para verificar este fato, basta substituir os pesos de  $G$  por seus respectivos logaritmos e identificar a MST do grafo  $G'$  resultante
- ★ Neste problema em particular, por conta das restrições dos pesos, use  $\log_2()$
- ★ O produto mínimo  $p$  da MST original pode ser obtido a partir do custo  $k$  da MST de  $G'$  por meio da expressão  $p = 2^k$

## Solução

- ★ A MST minimiza também o produto dos pesos das arestas
- ★ Para verificar este fato, basta substituir os pesos de  $G$  por seus respectivos logaritmos e identificar a MST do grafo  $G'$  resultante
- ★ Neste problema em particular, por conta das restrições dos pesos, use  $\log_2()$
- ★ O produto mínimo  $p$  da MST original pode ser obtido a partir do custo  $k$  da MST de  $G'$  por meio da expressão  $p = 2^k$
- ★ O número de divisores de  $2^k$  é igual a  $k + 1$

```
int solve(int N)
{
    auto cost = prim(1, N);

    return cost + 1;
}

int log2int(long long x)
{
    int res = 0;

    while (x > 1)
    {
        ++res;
        x >>= 1;
    }

    return res;
}
```