

## Descrição do problem B : Bacalhaus Congelados

Prova em curso

## Bacalhaus congelados

No início de dezembro de 2012, trinta bacalhaus oriundos da Noruega com destino ao primeiro aquário de bacalhaus do país em Ílhavo chegaram congelados. Segundo fonte noticiosa (tvi24), "os peixes, com cerca de 30 centímetros e 1,5 quilos, chegaram mortos, envoltos por blocos de gelo, ao que tudo indica devido a negligência no transporte, uma longa viagem desde Alesund até Ílhavo, entre avião e transporte rodoviário". É possível que tenham sido sujeitos a temperaturas de 50 graus negativos no transporte aéreo.

A propósito deste caso concreto, interessa-nos considerar uma rede de transportes com informação sobre a temperatura esperada em cada troço e determinar, para percursos indicados, qual a temperatura mínima e máxima esperada.

### Input

Na primeira linha tem o número de nós da rede e o número de troços (ramos) da rede.

A seguir tem a descrição dos troços da rede. Cada troço é descrito numa linha por quatro inteiros separados por espaços: os dois primeiros identificam os extremos do troço, o seguinte a temperatura esperada nesse troço e o último um custo associado ao transporte (que não é muito relevante). Qualquer ligação é bidirecional. Pode haver nós isolados. Os nós são identificados por inteiros consecutivos a partir de 1. Se for útil, pode assumir que a rede não tem mais do que 20000 nós.

Por fim tem a informação sobre os percursos. Terá sempre a descrição de pelo menos um percurso. Cada percurso é descrito numa linha, e é definido pela sequência de nós correspondente. O primeiro inteiro nessa linha é o número de nós na sequência. Deve assumir que todos os percursos dados são válidos, sendo o primeiro nó indicado a origem do percurso e o último o seu fim. Um percurso pode conter nós repetidos mas não em posições consecutivas. Os dados terminam por 0.

### Output

Para cada percurso, terá uma linha com a temperatura mínima e máxima correspondente, separadas por um espaço.

### Exemplo 1

#### Input

```
6 10
1 4 10 35
3 5 -22 30
1 2 -27 50
5 2 20 51
3 2 3 27
2 4 -1 5
5 6 9 20
1 5 30 20
4 6 25 450
6 3 -15 40
9 5 1 4 2 3 6 4 2 1
5 1 4 2 3 6
2 1 5
5 2 3 6 4 2
7 5 6 4 1 5 2 3
0
```

#### Output

```
-27 30
-15 10
30 30
-15 25
3 30
```

