

# MC558A - Projeto e Análise de Algoritmos II

Lehilton Pedrosa  
Murilo de Lima (PED C)

---

## Exercício de Programação II

- **Prazo de submissão:** 16 de abril às 23:59:59
- O exercício deve ser implementado em C ou C++
- Número máximo de submissões: 10
- Tempo máximo de execução: 10s

## Ferrovias

Milda é a presidenta de um determinado país B. Esse país é dividido em estados, e cada estado possui uma capital.

Milda quer reestruturar o sistema de estradas e ferrovias e precisa da sua ajuda. As ferrovias são muito antigas, e seu custo de manutenção é alto. Seu trabalho é ajudar a presidenta a decidir quais ferrovias podem ser desativadas. No entanto, como B é um país democrático, uma ferrovia só pode ser desativada se isso não piorar a qualidade do sistema de transporte.

Considere que o país B tem  $n$  cidades numeradas de 1 a  $n$ , das quais  $1 \leq k \leq n$  são capitais. O país tem  $m$  estradas e  $p$  ferrovias. Se uma estrada ou ferrovia liga a cidade  $i$  à cidade  $j$ , então você pode utilizá-la para ir tanto de  $i$  para  $j$  como de  $j$  para  $i$ . No entanto, uma das pontas de uma ferrovia sempre é uma capital. Cada estrada e ferrovia possui um comprimento; se entre o mesmo par de cidades houver uma estrada e uma ferrovia, elas podem ter comprimentos diferentes. Uma ferrovia pode ser desativada apenas se a distância de cada cidade que não é capital à capital mais próxima não for modificada. Seu objetivo é encontrar o número máximo de ferrovias que podem ser desativadas.

**Entrada:** na primeira linha da entrada são dados três inteiros  $n$ ,  $m$  e  $p$ , que indicam o número de cidades, de estradas e de ferrovias. Você pode supor que  $2 \leq n \leq 10^5$ , que  $1 \leq m \leq 3 * 10^5$  e que  $1 \leq p \leq 10^5$ . A seguir são dadas  $m$  linhas, descrevendo as estradas. Cada linha contém três números  $i$ ,  $j$  e  $c$ , com  $i \neq j$  e  $1 \leq c \leq 10^3$ , indicando que existe uma estrada de  $i$  a  $j$  com comprimento  $c$ .

A seguir são dadas  $p$  linhas, descrevendo as rodovias. Cada linha contém três números  $i$ ,  $j$  e  $c$ , com  $i \neq j$  e  $1 \leq c \leq 10^3$ , indicando que  $i$  é uma capital, e que existe uma ferrovia de comprimento  $c$  entre  $i$  e  $j$ .

Você pode supor que, para cada cidade, existe um caminho até pelo menos uma capital.

Você pode supor que existe no máximo uma estrada e no máximo uma ferrovia entre cada par de cidades.

Pode haver ferrovias de mesmo comprimento ligando uma cidade a mais de uma capital.

**Saída:** uma única linha com o número de ferrovias que podem ser desativadas.

### Exemplos:

Entrada:

5 5 3

1 2 1

2 3 2

1 3 3

3 4 4

1 5 5

1 3 5

1 4 5

1 5 5

Saída:

2

Entrada:

5 2 2

2 4 2

4 5 2

1 3 1

2 4 3

Saída:

1

**Relatório:** você deve incluir, no cabeçalho do arquivo do seu código, um comentário com 100 a 300 palavras, explicando sua solução. Não é necessário fazer uma prova formal, mas você deve argumentar por que sua solução funciona.

### Dicas:

- pense em como resolver o problema supondo que há apenas uma capital;
- depois pense em como solucionar o problema geral, supondo que há uma capital federal interligando as capitais estaduais.

### Observações:

- O SuSy utiliza o GCC 4.4.7 20120313 (Red Hat 4.4.7-17). São utilizadas as seguintes flags para compilação:
    - C99: `-std=c99 -pedantic -Wall -lm`
    - ANSI C: `-ansi -pedantic -Wall -lm`
    - C++: `-ansi -pedantic -Wall -lm`
  - Você deve implementar estruturas de dados eficientes, com consumo de memória  $O(n + m + p)$ .
  - Seu algoritmo deve executar em tempo  $O((n + m + p) \log n)$ . Lembre-se que alocação de memória (mesmo memória estática alocada na pilha) influi na complexidade de tempo.
  - A nota do exercício é proporcional ao número de casos de teste que você acertar; são dados 10 casos de teste, sendo 7 abertos e 3 fechados, valendo 1 ponto cada.
  - No entanto, **implementações com complexidade de memória ou tempo fora do especificado receberão nota zero no exercício.** Isto poderá ser verificado através de casos de teste fechados adicionais, com tamanho de entrada maior, executados fora do SuSy pelo monitor.
  - Você pode utilizar as bibliotecas-padrão do C e as estruturas de dados da biblioteca-padrão do C++.
  - **Trechos de código copiados da Internet ou dx coleguinha configuram plágio.**
  - Sugerimos que você use `scanf` para fazer a leitura da entrada, a fim de garantir que seu código execute no tempo especificado.
  - Seu código deve estar identado, modularizado e bem comentado. Identifique-se e deixe claro quais estruturas de dados e algoritmos foram utilizados.
-