

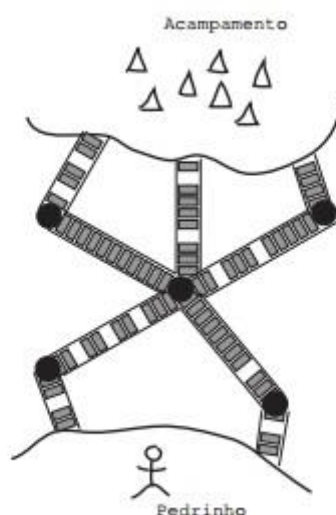
III BXComp

3º Campeonato de Programação para Calouros do Curso de Sistemas de Informação 2013

4ª Etapa – Desafio 1

Caminho das Pontes

Pedrinho é um rapaz muito aventureiro, que nas férias viaja pelo mundo em busca de lugares afastados e com vistas bonitas. Na sua viagem atual, Pedrinho está andando por uma escura floresta quando se depara com um perigoso desfiladeiro. Do outro lado do desfiladeiro ele sabe que existe um acampamento onde poderá descansar durante a noite para continuar suas aventuras no dia seguinte. Para chegar até o acampamento, ele terá que utilizar pontes que estão suspensas sobre o desfiladeiro. As pontes foram construídas interligando altos pilares cravados no fundo do desfiladeiro.



O piso das pontes é feito de tábuas de tamanhos iguais. Mas as pontes são velhas, e algumas tábuas caíram. Felizmente, todas as tábuas que sobraram estão em perfeitas condições, ou seja, não existe perigo de Pedrinho pisar em uma delas e a tábua cair. Além disso, em nenhuma das pontes duas tábuas consecutivas caíram, de forma que os buracos deixados pelas tábuas que caíram podem ser pulados com segurança.

No local onde Pedrinho se encontra existe uma placa mostrando as ligações entre as pontes e também quantas tábuas estão faltando em cada uma das pontes. Pedrinho está cansado, com pressa para descansar, e não há muita visibilidade durante a noite. Ele precisa, no entanto, tomar muito cuidado para não cair em algum dos buracos.

Pedrinho possui um laptop na mochila, mas só o usa para comunicar-se com os amigos. Ele liga sua internet via satélite, encontra você on-line, e pede sua ajuda.

Tarefa

Sua tarefa é escrever um programa que receba as informações sobre as pontes (as ligações entre elas e a quantidade de tábuas faltando em cada uma) e **sempre** escolha a ponte que, dentre as opções possíveis durante a travessia, é aquela que possui o menor número de tábuas faltando, ou seja, a ponte a ser escolhida deve ser **sempre** a mais promissora **no momento** (aquela que terá menor número de buracos para pular) pra chegar na próxima ponte a ser atravessada.

Repare que escolher a ponte **mais promissora no momento**, em boa parte das vezes realmente fará com que Pedrinho passe pelo menor número de buracos possível em toda a travessia. Mas existem casos nos quais escolher a opção que parece ser a melhor no momento pode implicar em um caminho que não seria o melhor para a travessia como um todo. Ou seja, a estratégia que você **deve** usar pode fazer com que Pedrinho passe por um caminho com mais tábuas faltando do que seria necessário.

Entrada

A entrada será composta por vários casos de teste; a primeira linha possuirá um número natural que indica a quantidade de casos de teste.

Cada caso de teste terá dois números inteiros na primeira linha, N e M , representando o número de pilares no desfiladeiro e o número de pontes, respectivamente. Assim, cada caso de teste contém M linhas em sequência (ou seja, de acordo com o número de pontes) compostas por 3 inteiros, nesta descrição representados como S , T , B , indicando que existe uma ponte ligando os pilares S e T , e que essa ponte possui B buracos. Não existe linha representando ponte com $S = T$. O valor de pilar 0 representa a borda do desfiladeiro onde Pedrinho está, e o valor de pilar $N + 1$ representa a borda do desfiladeiro onde está o acampamento. Não existem duas pontes distintas ligando o mesmo par de locais (pilares/bordas do desfiladeiro), ou seja, não existem duas pontes que liguem exatamente os mesmos dois pilares.

Você pode supor que sempre existirá um caminho de pontes entre o lado do desfiladeiro em que Pedrinho se encontra até o lado do desfiladeiro onde está o acampamento. E também que pontes que saem do mesmo pilar (ou da borda do desfiladeiro) não possuem o mesmo número de buracos.

Saída

Seu programa deve imprimir, na saída padrão, uma frase (veja o *exemplo de saída*) contendo a quantidade de buracos que Pedrinho terá que pular para conseguir chegar ao acampamento seguindo o caminho que pareceu ser o mais vantajoso.

Exemplo de entrada

3

2 5

0 1 1

0 2 3

0 3 9

1 3 2

2 3 2

4 9

0 1 1

0 3 4

0 4 2

1 2 5

1 5 3

2 5 5

3 4 2

3 5 5

4 5 8

4 9

0 1 1

0 3 4

0 4 2

1 2 2

1 5 3

2 5 5

3 4 2

3 5 5

4 5 8

Exemplo de Saída

Ceguei no acampamento passando por 3 buracos!

Ceguei no acampamento passando por 4 buracos!

Ceguei no acampamento passando por 8 buracos!