

PxNP

Trabalho 1 de MC458 - Entrega: 14/05

Um parque famoso pela sua grande quantidade de montes e pelas belas vistas que estes proporcionam é o Parque exótico Nacional dos Picos (PxNP). Uma das atividades que são realizadas no parque é uma excursão pelos montes para apreciar as paisagens. Para tornar a excursão mais interessante e prática, os montes são interligados por tirolesas. Assim, para cada par de montes, existe exatamente uma tirolesa que os conecta, que logicamente só pode ser usada em um sentido.

Você ficou responsável por determinar uma rota para a excursão no parque. A excursão pode começar por qualquer monte, e não se deseja passar por um mesmo monte mais de uma vez. O PxNP é composto por um total de N montes, e você é capaz de consultar se há uma tirolesa que conecta um monte ao outro. Sua tarefa é determinar uma sequência de montes que podem ser visitados de forma a visitar o maior número de montes possível.

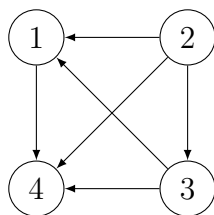
Entrada e Saída

É dado um inteiro N que representa o número de montes. Os montes são rotulados com inteiros de 1 a N . Também é fornecido uma função `has_edge(i,j)` que informa se há uma tirolesa que conecta os montes i e j , nesse sentido.

Você deve escrever uma rotina que devolve um vetor que representa uma sequência de montes para ser visitados de tamanho máximo. Caso haja mais de uma sequência válida, você pode devolver qualquer uma.

Exemplo

Considere o seguinte exemplo com $N = 4$ montes:



Neste caso existe uma rota que passa pelos 4 montes, dada pela sequência 2, 3, 1, 4.

Restrições

- 50% dos casos de teste: $1 \leq N \leq 10^3$;
- 50% dos casos de teste: $1 \leq N \leq 10^5$;
- Tempo limite de execução: 1 segundo.

Sua nota será dada pela quantidade de casos de teste corretos e dentro do tempo limite que seu algoritmo é capaz de resolver. Note que os testes não vão apenas verificar corretude, mas também a complexidade de tempo para resolver o teste. Assim, programas corretos mas com complexidade de tempo alta provavelmente não resolverão a tempo os testes maiores. Para obter uma estimativa da complexidade de tempo esperada, pode-se estimar que a máquina é capaz de realizar até uma quantidade da ordem de 10^7 instruções dentro de 1s. Com isso, note que um algoritmo com complexidade $\mathcal{O}(N^2)$ não deve passar os casos de teste maiores.

Instruções para submissão

- Você deve entregar um código `.c` ou `.cpp`, além de um relatório que explica seu algoritmo.
- A entrega do código será feita pelo SuSy, cujo link será informado em breve.
- O código pode ser escrito em C ou C++. Serão usados os parâmetros de compilação `-std=c99 -pedantic -Wall -lm` para C, e `-ansi -pedantic -Wall -lm` para C++.
- É permitido um total de 20 submissões. **Apenas a submissão mais recente será considerada.**
- A entrega do relatório será feito pelo Classroom, na tarefa correspondente, através de um documento em pdf. O relatório deve contemplar a descrição do algoritmo, bem como a **análise da complexidade de tempo** do algoritmo implementado e de eventuais estruturas de dados usadas.