

MC558 - Complexidade de Algoritmos II

Primeiro semestre de 2025

Laboratório 5

Um mundo de aventuras

Parabéns! Ou... sinto muito!? Você foi escolhida para testar uma variedade de jogos do tipo *adventure*, com a missão de descobrir quais deles podem ser completados com sucesso.

Cada jogo é composto por um conjunto de $n \leq 100$ salas. Uma dessas salas é a **sala inicial**, e outra é a **sala final**. Cada sala possui um **valor de energia**, que varia entre -100 e 100 . Portas unidirecionais conectam pares de salas; ou seja, a jogadora pode usar uma porta para ir de uma sala a outra, mas não no sentido inverso. Nossa jogadora começa na sala inicial com 100 pontos de energia. Ela pode atravessar qualquer porta que conecte a sala em que está a outra sala, entrando assim na próxima sala. O valor de energia desta nova sala é então adicionado à energia da jogadora. Esse processo continua até que ela vença o jogo — ao entrar na sala final — ou morra por falta de energia (ou desista por frustração...). É importante observar que, para não morrer por falta de energia, a jogadora deve manter sua energia estritamente positiva durante todo o percurso. Durante sua jornada, a jogadora pode entrar na mesma sala várias vezes, recebendo o bônus de energia da sala a cada visita.

Observação: Não reinvente a roda! Tente utilizar os algoritmos vistos em aula para ajudar a resolver este problema.

1 Entrada e Saída

Entrada: A primeira linha da entrada contém um inteiro n , que indica o número de salas do jogo. Considere que as salas estão rotuladas de 0 até $n - 1$, sendo a sala inicial sempre a sala 0, e a sala final, a sala $n - 1$. A segunda linha da entrada contém n inteiros w_0, \dots, w_{n-1} , onde w_i representa a energia da sala i , com $-100 \leq w_i \leq 100$. A sala inicial e a sala final possuem sempre energia 0, ou seja, $w_0 = w_{n-1} = 0$. A terceira linha da entrada contém um inteiro m , indicando o número de portas do jogo. As próximas m linhas contêm, cada uma, dois inteiros u e v , com $0 \leq u, v \leq n - 1$, indicando que há uma porta que permite sair da sala u e entrar na sala v .

Saída: Imprima **possible** se for possível vencer o jogo, ou **impossible** caso contrário (todas as letras em minúsculas e com quebra de linha ao final).

2 Exemplos

Entrada	Saída
5 0 -60 -60 20 0 4 0 1 1 2 2 3 3 4	impossible
5 0 20 -60 -60 0 4 0 1 1 2 2 3 3 4	impossible
5 0 21 -60 -60 0 4 0 1 1 2 2 3 3 4	possible
5 0 20 20 -60 0 5 0 1 1 2 2 1 2 3 3 4	possible

3 Implementação e Submissão

- A solução deverá ser implementada em C, C++ ou Python 3.
- O programa deve ser submetido no SuSy, com o nome principal **t5** (por exemplo, t5.c).
- O número máximo de submissões é 20.
- A tarefa contém 10 testes abertos e 10 testes fechados. A nota será proporcional ao número de acertos nos testes fechados.
- **Casos de plágio implicam em nota ZERO na disciplina para todos os envolvidos.**
- **Não é permitido o uso de bibliotecas que não sejam padrão, bem como diretivas ou flags de otimização.**

4 Prazo final de submissão

Segunda-feira, dia 2 de Junho, às 6h da manhã.