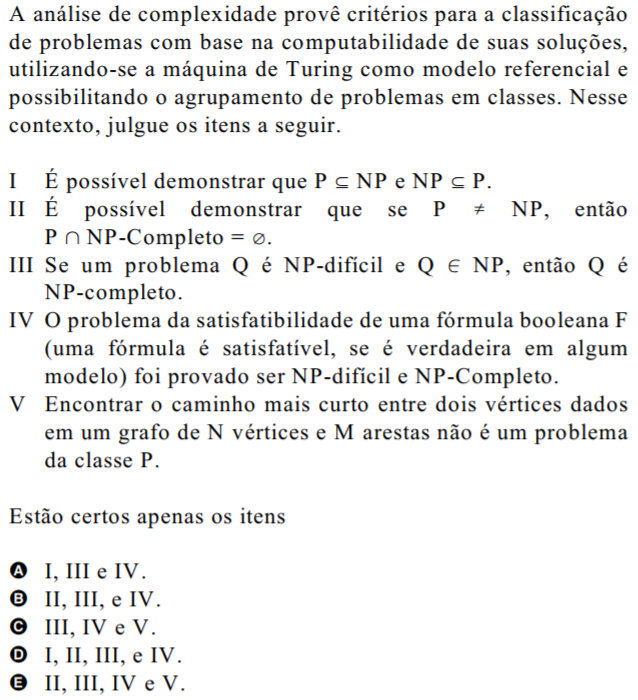
Lista de Exercícios - NP Completude

1. Explique, com suas palavras, o que são os conjuntos P, NP, NP-Completo e NP-Difícil.
2. Imagine que existe um problema X e não se sabe se o mesmo é NP-Completo. Quais requisitos ele precisa atender para ser NP-Completo?
3. O que é redução polinomial? Quais passos devem ser realizados nesse procedimento?
4. Como é possível mostrar que um problema é NP? Apresente 3 exemplos para problemas distintos.
5. Problemas NP-Dificil são redutíveis polinomialmente?
6. É possível afirmar que P está contido em NP?
7. É possível afirmar que NP está contido em P?
8. Todo problema NP-Completo é NP-Difícil?
9. Todo problema NP-Difícil é NP-Completo?
10. É possível tornar um problema de otimização em decisão? Como?
11. Por que o problema de ciclo euleriano é P e ciclo hamiltoniano é NP-Completo? Explique com suas palavras.
12. Porque o problema do caminho simples em grafos é P e caminho simples de tamanho máximo em grafos é NP-Completo? Explique com suas palavras.
13. Por que o problema de 2-satisfabilidade é P e 3-Satisfabilidade é NP-Completo? Explique com suas palavras.
14. Qual foi o primeiro problema NP-Completo conhecido? Como ele surgiu?
15. O jogo Tetris é P, NP-Completo ou NP-Difícil? Justifique.



1. O problema P versus NP é um problema ainda não resolvido e um dos mais estudados em Computação. Em linhas gerais, deseja-se saber se todo problema cuja solução pode ser eficientemente verificada por um computador, também pode ser eficientemente obtida por um computador. Por “eficientemente” ou “eficiente” significa “em tempo polinomial”. A classe dos problemas cujas soluções podem ser eficientemente obtidas por um computador é chamada de classe P. Os algoritmos que solucionam os problemas dessa classe têm complexidade de pior caso polinomial no tamanho das suas entradas. Para alguns problemas computacionais, não se conhece solução eficiente, isto é, não se conhece algoritmo eficiente para resolvê-los. No entanto, se para uma dada solução de um problema é possível verificá-la eficientemente, então o problema é dito estar em NP. Dessa forma, a classe de problemas para os quais suas soluções podem ser eficientemente verificadas é chamada de classe NP. Um problema é dito ser NP-completo se pertence à classe NP e, além disso, se qualquer outro problema na classe NP pode ser eficientemente transformado nesse problema. Essa transformação eficiente envolve as entradas e saídas dos problemas. Considerando as noções de complexidade computacional apresentadas acima, analise as afirmações que se seguem.

I. Existem problemas na classe P que não estão na classe NP.

II. Se o problema A pode ser eficientemente transformado no problema B e B está na classe P, então A está na classe P.

III. Se P = NP, então um problema NP-completo pode ser solucionado eficientemente.

IV. Se P é diferente de NP, então existem problemas na classe P que são NP-completos. É correto apenas o que se afirma em

A) I.

B) IV.

C) I e III.

D) II e III.

E) II e IV.

1. Analise as seguintes afirmativas.

I. Em um problema de decisão, o objetivo é decidir a resposta sim ou não a uma questão. Em um problema de localização, procura-se localizar uma certa estrutura que satisfaça um conjunto de propriedades dadas. Se as propriedades envolverem critérios de otimização, então o problema é dito de otimização.

II. A teoria da complexidade restringe-se a problemas de decisão, já que o estudo de problemas NP-completos é aplicado somente para esse tipo de problema.

III. Os problemas NP-Completos são considerados como os problemas mais difíceis em NP. Se qualquer problema NP-Completo pode ser resolvido em tempo polinomial, então todos os problemas em NP podem ser resolvidos da mesma forma.

A análise permite concluir que

A) apenas a afirmativa I está correta.

B) apenas a afirmativa II está correta.

C) apenas as afirmativas I e II estão corretas.

D) apenas as afirmativas I e III estão corretas.

E) todas as afirmativas estão corretas.