

Explique de forma sucinta a diferença entre problemas tratáveis, intratáveis e insolúveis. De exemplos de cada um deles.

Problemas tratáveis são aqueles que possuem algoritmos capazes de resolvê-los com complexidade polinomial (Exemplo: ordenação de um conjunto de inteiros, cálculo de matrizes, entre outros). Já algoritmos intratáveis são aqueles que nenhum algoritmo com tempos polinomiais podem resolvê-los. Podemos citar como exemplo o problema dos Ladrilhos de Wang. Por fim, os problemas insolúveis são aqueles que não são computáveis, ou seja, aqueles que não há computabilidade representada por uma Máquina de Turing Universal, um exemplo que o próprio Alan Turing apresentou foi o *problema da parada*.

Descreva as classes de complexidade P, NP, coNP e NP-completo. De a definição, estabeleça o relacionamento entre as classes e dê exemplos de cada uma delas.

A classe de complexidade algorítmica P são todos aqueles problemas que podem ser resolvidos por um algoritmo polinomial. Em união às classes de complexidade, estão os algoritmos NP, que possuem um verificador para a resposta SIM polinomial e seu inverso são os coNP, que possuem um algoritmo polinomial para a resposta negativa. Por fim, os problemas NP-completo se encaixam como um grupo de problemas NP com característica de poderem ser reduzidos para problemas com complexidade polinomial.



Responda $P = NP$? Justifique sua resposta!

Existe uma propriedade da complexidade de algoritmos que determina que se um problema NP-completo for resolvido em um tempo polinomial, então todo problema NP tem uma solução polinomial $P=NP$.