

P vs NP:

O principal problema aberto da Ciência da Computação

João Pedro Machado Silva, BV3032477

A Complexidade Computacional é um nicho da Matemática Computacional que foca no estudo da eficiência de algoritmos. Basicamente, não importa se um algoritmo foi perfeitamente desenvolvido, se o tempo para que o mesmo seja executado seja relativamente longo e ao considerar que a multiplicação de números grandes milhares de vezes seja necessário para alguns algoritmos, o tempo passa a ser extremamente longo.

Para fazer um cálculo desse tempo que o programa leva da entrada dos dados até a resposta, usamos um tempo teórico, do qual é calculado definindo uma certa unidade de tempo para cada operação realizada. Se a dependência do tempo com relação aos dados de entrada for polinomial, trata-se de um programa rápido, já se a dependência do tempo for exponencial, trata-se de um programa lento. Além disso, tem-se os programas que estão nas duas classes.

Classe P: É uma classe em que os problemas podem ser resolvidos de forma eficiente em tempo polinomial por um algoritmo determinístico. Ou seja, o tempo de execução do algoritmo cresce de forma polinomial com o tamanho da entrada do problema. Resumindo, são os problemas que podem ser resolvidos em um tempo razoável, mesmo para entradas grandes.

Exemplos da Classe P:

- Ordenar uma lista, que consiste em uma lista de números, que deverão ser ordenados de modo crescente ou decrescente.
- Determinar se um número é primo, que consiste em verificar se um número é primo ou não.
- Busca em lista ordenada, que consiste em buscar a posição de um número em uma lista já ordenada.

Classe NP: Classe em que os problemas são aqueles para os quais uma solução pode ser verificada em tempo polinomial, embora a solução em si possa exigir um tempo exponencial para ser encontrada. Ou seja, se alguém apresentar uma solução para um problema NP, você pode verificar facilmente se a solução está correta em tempo polinomial, mas encontrar essa solução pode ser muito difícil.

Exemplos da Classe NP:

- O problema do caixeiro viajante(Traveling Salesman Problem), consiste em encontrar a rota mais curta entre algumas cidades, passando uma vez em cada uma delas e retornando à cidade de origem.
- O problema da mochila(Knapsack Problem), consiste em combinar itens dados, cada um com um peso e um valor, de forma a maximizar o valor total, enquanto tem-se que manter o peso total abaixo de um limite dado.

Concluindo, é importante lembrar que um problema da classe NP não é “mais difícil” que os problemas da classe P, pois na verdade, não se sabe se existem problemas NP que não estão inclusos na classe P, ou seja, não se sabe se P é igual a NP ou não. Esta questão está em aberto na teoria da computação e é nomeada “Conjectura P versus NP”.

Referências:

- “P vs NP, o problema do milênio”, ideias esquecidas, disponível em: <https://ideiasesquecidas.com/2023/04/01/p-vs-np-o-problema-do-milenio/>
- “P versus NP”, ufscar, disponível em: <https://www.dm.ufscar.br/hp/hp501/hp501001/hp501001.html>
- “Complexidade e problemas NP-completos”, ime usp, disponível em: https://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/NPcompleto.html