

Sobre Invariantes

Por Prof. M.Sc. Rodrigo Hagstrom

São Paulo, Janeiro de 2025

Invariantes em análise de algoritmos referem-se a propriedades ou condições que permanecem verdadeiras durante toda a execução de um algoritmo, independentemente do estado atual da execução. Esses conceitos são frequentemente usados para demonstrar a correção de algoritmos, pois ajudam a provar formalmente que um algoritmo alcançará o resultado desejado.

Um invariante é geralmente definido no início da execução e deve ser mantido após cada iteração ou passo do algoritmo. No final, a verificação do invariante pode garantir que o algoritmo produziu o resultado correto.

Por exemplo, no algoritmo de ordenação Insertion Sort, o invariante é que a parte do array à esquerda do índice atual está sempre ordenada. Cada iteração expande essa parte ordenada ao inserir o próximo elemento na posição correta, mantendo o invariante.

Importância dos Invariantes:

- **Prova de Correção:** Demonstram formalmente que o algoritmo faz o que se propõe a fazer.
- **Compreensão do Algoritmo:** Ajudam a entender o funcionamento interno do algoritmo.
- **Desenvolvimento de Algoritmos:** Guiam a formulação e o design de algoritmos corretos.

Para validar um invariante, geralmente são usados três passos:

1. **Inicialização:** Mostrar que o invariante é verdadeiro antes da primeira iteração.
2. **Manutenção:** Mostrar que, se o invariante for verdadeiro antes de uma iteração, ele continuará verdadeiro depois.
3. **Término:** Mostrar que o invariante implica a correção do resultado quando o algoritmo termina.