

Comentário Sobre as Buscas:

1. Busca Sequencial:

A vantagem da busca sequencial se dá na simplicidade de ser implementada em código, pois ela é executada de maneira semelhante à que um cérebro humano faria sem o auxílio de um computador. Entretanto, ela não é a mais recomendada para ordenar vetores com quantidades grandes de elementos, pois assim, se o elemento buscado estiver em uma das últimas posições do vetor ou se o mesmo elemento não se encontrar no vetor, o algoritmo terá que percorrer quase todo (ou todo) o vetor para retornar a chave buscada, o que leva um grande tempo de execução.

O Melhor caso da busca sequencial se dá quando a chave buscada se encontra na primeira posição do vetor, pois, nessa situação, faz a primeira verificação e já retorna. Os piores casos, como citado acima, ocorrem quando a chave buscada se encontra na última posição do vetor ou quando a chave não é encontrada no vetor.

2. Buscas Binárias:

As buscas binárias, tanto a iterativa quanto a recursiva, apesar de exigirem mais linhas de código, são bem mais eficientes que a busca sequencial, pois vão “quebrando” o tamanho do vetor à metade a cada verificação e, assim, mesmo com vetores com quantidades enormes de elementos, o baixo tempo de execução é notado.

A diferença entre as buscas binárias iterativa e recursiva se dá na maneira de implementação. A maneira recursiva, pessoalmente, é mais fácil de ser implementada, mas, ao mesmo tempo, gasta uma quantidade maior de memória Stack durante seu tempo de execução.

O melhor caso da busca binária ocorre quando a chave buscada está exatamente na posição central do vetor, visto que essa posição é a primeira a ser verificada. Já os piores casos ocorrem quando a chave buscada se encontra em uma das extremidades do vetor (primeira e última posição) ou quando a chave não é encontrada.