

Exercício 1

A complexidade temporal de busca no pior caso para esse algoritmo é $O(n)$.

Supondo que o vetor tenha n elementos, no pior caso o algoritmo precisará percorrer todos os elementos até concluir que o elemento foi ou não encontrado, por isso a complexidade é de $O(n)$.

A complexidade no pior caso da versão recursiva é a mesma que a versão iterativa.

Exercício 2

A complexidade temporal de busca no pior caso para esse algoritmo é $O(\log n)$.

A complexidade espacial é $O(1)$ para a versão iterativa, pois não há chamada de função adicionais. Enquanto para a versão recursiva é $O(\log n)$ devido a outras chamadas de função que crescem conforme o número de chamadas recursivas.

Exercício 3

A complexidade temporal de busca tanto no pior quanto no melhor caso para esse algoritmo é $O(n)$, pois em ambos os casos é necessário percorrer o vetor uma vez.

Exercício 4

A complexidade temporal de busca no pior caso para esse algoritmo é $O(N)$, onde N é o número da posição desejada.

Em comparação a versão recursiva, a complexidade temporal no pior caso na recursiva é $O(2^N)$, pois a versão recursiva tem uma chamada recursiva para cada número da sequência até chegar ao número desejado.