

Atividade 1 - Prática

Entrega: 24/11/2020

Análise de Complexidade do Algoritmo -

Analisando o código criado, vislumbramos uma função iterativa e uma função recursiva que possui a mesma lógica: Ela toma o valor primo, itera todos os valores até chegar no valor primo, e caso não seja divisível em nenhuma iteração, ele retorna que o valor é primo e caso seja divisível, ele retorna 0 direto, interrompendo o laço. O range de números aleatórios colocado foi de 0 a 10000, ou seja, o último primo dado é 9973, podendo o código alcançar até esse valor no pior dos casos, assim como pode alcançar o valor 1, retornando na mesma hora.

```
O maior valor(Iterativo) é 9941
O algoritmo iterativo demorou 0.003846 segundos para executar
O maior valor(Recursivo) é 9941
O algoritmo recursivo demorou 0.011802 segundos para executar
```

Figura 1 - A saída do código rodado em uma das vezes, onde o valor maior foi 9941 (Fonte: Autor)

Para a função iterativa, houve, quando foi rodado o vetor aleatório de 1000 números, um valor de 0.003846 segundos, ou seja, uma velocidade interessante até certo ponto, visto que são diversas iterações, considerando que o pior caso foi o de 9941, ou seja, 9941 iterações para analisar em tão somente um valor.

Para a função recursiva, houve, quando foi rodado o vetor aleatório de 1000 números, um valor de 0.011802 segundos, ou seja, uma velocidade interessante até certo ponto, mesmo que o tempo tenha sido bem maior do que a da função iterativa. Uma função recursiva, matematicamente, é executada duas vezes num mesmo código, visto que ele precisa iterar n vezes e depois retornar também n vezes. Considerando que o pior caso foi o de 9941, temos uma relação entre o tempo iterativo e o tempo recursivo, sendo o tempo recursivo bem maior por conta dos grandes valores que foram iterados por vezes na função recursiva.