

Aula 01 - Exercício prático análise de complexidade

Submeter o código em arquivo .c/.py/.java. O nome do arquivo deve ser o nome_sobrenome_aula1Pratico

Submeter o tempo de execução e a análise de complexidade num arquivo pdf. O nome do arquivo deve ser o nome_sobrenome_aula1Pratico

1) (1,0) Proponha um algoritmo recursivo para determinar se um número é primo ou não. Proponha um algoritmo iterativo para o mesmo fim.

a) Execute ambas as funções para encontrar o maior número primo num vetor aleatório de 100.000 elementos (use o mesmo vetor em ambos os códigos).

b) Verifique qual dos dois executou mais rápido (use uma time() function).

Resposta: Para um vetor aleatório de 100.000 elementos com cada elemento sendo um número de 1 a 100000. Foram obtidos os seguintes resultados:

```
aaed git:(main) ✗ /opt/homebrew/bin/python3.10 /Users/cerka/projects/pos/aaed/aula01/praticos/primos.py
Biggest prime number using iterative method: 99989, time: 0.06051206588745117
Biggest prime number using recursive method: 99989, time: 0.34767818450927734
```

O método que executou mais rápido foi o iterativo.

c) Faça a análise de complexidade de tempo de ambos os algoritmos.

A função iterativa possui uma atribuição de variável na linha 6 e uma verificação condicional na linha 8, ambas as operações possuem tempo constante $O(1)$. Portanto, a operação de laço FOR na linha 11 é a operação que acarreta maior custo computacional. A laço é percorrido de 2 até \sqrt{n} vezes. Como a operação condicional interna do laço tem tempo constante $O(1)$. A complexidade de tempo do algoritmo iterativo é $O(\sqrt{n})$.

A função recursiva possui 3 operações condicionais nas linhas 19, 22 e 25 todas possuem tempo constante $O(1)$. Na linha 28 é feita a chamada para a função dando início a recursão. A função termina a execução quando é encontrado um divisor diferente de 1 ou o próprio número, com resultado não primo ou quando o divisor torna-se maior que \sqrt{n} . Portanto, no pior dos casos esse algoritmo possui tempo de execução $O(\sqrt{n})$.