

- 1) Fazer um pequeno texto mostrando qual é a diferença entre as notações  $O$ ,  $\Omega$  e  $\Theta$ ?

Notação Big  $O$ : É o pior caso de execução, por exemplo, se quiser pesquisar um item em uma lista de  $n$  itens, o pior caso será se o item procurado estiver na última posição ou não existir no *array* em questão. Sendo que terá uma complexidade  $n$ .

Notação Big  $\Omega$ : É o melhor caso de execução, por exemplo, se quiser pesquisar um item em uma lista de  $n$  itens, o melhor caso será se o item procurado estiver na primeira posição da lista, sendo assim, lerá a primeira posição e já retornará o mesmo.

Notação Big  $\Theta$ : É o caso médio de execução, em grosso modo, pode ser obtido com a média de complexidade entre o melhor e o pior caso, como citado de exemplo, se o item procurado estivesse no meio da lista, a complexidade seria de  $n/2$ .

- 2) Para cada exercício da Unidade 01b - Noções de complexidade, apresente sua ordem de complexidade usando as notações  $O$ ,  $\Omega$  e  $\Theta$ .

5)  $O: 1$                        $\Omega: 1$                        $\Theta: 1$

```
int i = 10;

while (i >= 7){
    i--;
}
```

6)  $O: 1$                        $\Omega: 1$                        $\Theta: 1$

```
...
for (int i = 5; i >= 2; i--){
    a--;
}
```

7)  $O: 1$

$\Omega: 1$

$\Theta: 1$

```
...  
for (int i = 0; i < 5; i++){  
    if (i % 2 == 0){  
        a--;  
        b--;  
    } else {  
        c--;  
    }  
}
```

8)  $O: n^2$

$\Omega: n^2$

$\Theta: n^2$

```
...  
for (int i = 0; i < n; i++){  
    for (int j = 0; j < n; j++){  
        a--;  
    }  
}
```

9)  $O: 1$

$\Omega: 1$

$\Theta: 1$

```
int i = 1, b = 10;  
  
while (i > 0){  
    b--;  
    i = i >> 1;  
}  
  
i = 0;  
  
while (i < 15){  
    b--;  
    i += 2;  
}
```

10)  $O: n^2$       $\Omega: n^2$       $\Theta: n^2$

```
for (int i = 0; i < n; i++)  
    for (int j = 0; j < n - 3; j++)  
        a *= 2;
```

11)  $O: n^2$       $\Omega: n^2$       $\Theta: n^2$

```
for (int i = n - 7; i >= 1; i--)  
    for (int j = 0; j < n; j++)  
        a *= 2;
```

12)  $O: \log(n)$       $\Omega: \log(n)$       $\Theta: \log(n)$

```
for (int i = n; i > 0; i /= 2)  
    a *= 2;
```

13)  $O: \log(n)$       $\Omega: \log(n)$       $\Theta: \log(n)$

```
for (int i = n+4; i > 0; i >>= 1)  
    a *= 2;
```

14)  $O: n^2$       $\Omega: n^2$       $\Theta: n^2$

```
for (int i = n - 7; i >= 1; i--)  
    for (int j = n - 7; j >= 1; j--)  
        a *= 2;
```

15)  $O: \log(n)$

$\Omega: \log(n)$

$\Theta: \log(n)$

```
for (int i = n + 1; i > 0; i /= 2)
    a *= 2;
```

16)  $O: \log(n)$

$\Omega: \log(n)$

$\Theta: \log(n)$

```
for (int i = n; i > 1; i /= 2)
    a *= 2;
```

17)  $O: \log(n)$

$\Omega: \log(n)$

$\Theta: \log(n)$

```
for (int i = 1; i < n; i *= 2)
    a *= 2;
```

18)  $O: \log(n)$

$\Omega: \log(n)$

$\Theta: \log(n)$

```
for (int i = 1; i <= n; i *= 2)
    a *= 2;
```