



## Unidade 1: Introdução - noções de complexidade

Exercício 1:  $2^x = y$

a)  $2^1 = 2$

c)  $2^2 = 4$

d)  $2^3 = 8$

e)  $2^4 = 16$

f)  $2^5 = 32$

g)  $2^6 = 64$

h)  $2^7 = 128$

i)  $2^8 = 256$

j)  $2^9 = 512$

k)  $2^{10} = 1024$

l)  $2^{11} = 2048$

2)  $\lg_2(2048) = 11$

b)  $\lg_2(4096) = 12$

c)  $\lg_2(512) = 9$

d)  $\lg_2(256) = 8$

e)  $\lg_2(128) = 7$

f)  $\lg_2(64) = 6$

g)  $\lg_2(32) = 5$

h)  $\lg_2(16) = 4$

i)  $\lg_2(8) = 3$

j)  $\lg_2(4) = 2$

k)  $\lg_2(2) = 1$

l)  $\lg_2(1) = 0$

3)  $\lceil \lg_2(16) \rceil = 4$

b)  $\lfloor \lg_2(16) \rfloor = 4$

c)  $\lceil \lg_2(17) \rceil = 5$

d)  $\lfloor \lg_2(17) \rfloor = 4$

e)  $\lceil \lg_2(15) \rceil = 4$

f)  $\lceil \lg_2(16) \rceil = 4$

g)  $\lceil \lg_2(17) \rceil = 5$

h)  $\lceil \lg_2(17) \rceil = 5$

i)  $\lceil \lg_2(17) \rceil = 5$

j)  $\lg_2(15) = 3,9069$

k)  $\lceil \lg_2(15) \rceil = 4$

l)  $\lfloor \lg_2(15) \rfloor = 3$

### Contagem de operações

Exercício Resolvido 1: Três subtrações

Exercício Resolvido 2: melhor caso: 3 adições

Pior caso: 5 adições

Exercício Resolvido 3: O número máximo de adições acontece quando a primeira condição do if é falsa e a segunda, verdadeira. Se a primeira condição for verdadeira, o Java não executa a segunda condição.

Exercício Resolvido 4: O código realiza 4 subtrações, quando i vale 0, 1, 2, 3.





0100 0101  
0010 0010  
0010 0110  
0011 0011

DISTOOS  
DLIMJVS

2<sup>o</sup> while: 0 < 15 V 8 < 15 V 16 < 15 X  
2 < 15 V 10 < 15 V  
4 < 15 V 12 < 15 V  
6 < 15 V 14 < 15 V

8 subtrações  
1 + 8 = 9 subtrações no total.

Exercício 10:  $n \cdot n - 3 \cdot 1 \rightarrow n^2 - 3$  multiplicações

Exercício 11:  $1 \cdot n \cdot 1 \rightarrow n$  multiplicações

Exercício 12: Quando  $n$  é uma potência de 2, realizamos  $\lg(n) + 1$  multiplicações  
Ex:  $n = 8$   
 $\lg 8 = 3 + 1 = 4 \rightarrow$  efetuamos a multiplicação quando  $i$  vale 8, 4, 2, 1

Para um valor qualquer de  $n$ , temos  $\lfloor \lg(n) \rfloor + 1$  multiplicações  
Ex:  $n = 7$   
 $\lfloor \lg(7) \rfloor + 1 = 2 + 1 = 3 \rightarrow$  efetuamos a multiplicação quando  $i$  vale 7, 3, 1

Exercício 13: efetuamos a multiplicação quando  $n \geq 4$

Exercício 14: efetuamos a multiplicação quando  $n \geq 7$

Exercício Resolvido 9: mesma resposta do exercício 12

☐  
☐  
☐



Exercício 15: Quando  $n$  é uma potência de 2, realizamos

$\lg(n) + 1$  multiplicações

$n \geq 0$

ex:  $n = 1$

$\lg(2) + 1 = 1 + 1 = 2 \rightarrow$  Efetuamos a multiplicação quando  $i$  vale 2, 1

Para um valor qualquer de  $n$ , temos  $\lfloor \lg(n) \rfloor + 1$  multiplicações

ex:  $n = 2$

$\lg(3) + 1 = 1 + 1 = 2 \rightarrow$  Efetuamos a multiplicação quando  $i$  vale 3, 1

Exercício 16:  $\lg(n) + 1$  multiplicações

$n \geq 1$

ex:  $n = 4$

$\lg(4) + 1 = 2 + 1 = 3$  multiplicações  $\rightarrow 4, 2, 1$

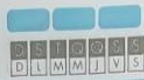
$\lfloor \lg(n) \rfloor + 1$

ex:  $n = 3$

$\lfloor \lg(3) \rfloor + 1 = 1 + 1 = 2$  multiplicações  $\rightarrow 3, 1$

Exercício 17:  $2n$  multiplicações

Exercício 18:  $2n$  multiplicações



### Exercício Resolvido 10 a)

$$3n + 2n^2$$

$i = 0;$

$\text{while } (i < n) \{$

$i++;$

$a--; b--; c--;$

$\}$

$\text{for } (i = 0; i < n; i++) \{$

$\text{for } (j = 0; j < n; j++) \{$

$a--; b--;$

$\}$

$\}$

$$b: 5n + 4n^3$$

$i = 0;$

$\text{while } (i < n) \{$

$i++;$

$a--; b--; c--; d--; e--;$

$\}$

$\text{for } (i = 0; i < n; i++) \{$

$\text{for } (j = 0; j < n; j++) \{$

$\text{for } (k = 0; k < n; k++) \{$

$a--; b--; c--; d--;$

$\}$

$\}$

$\}$

$\}$

$\}$

$\}$

$\}$





c)  $\lg(n) + n$

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

```
    a--;
```

```
for (int i = 0; i < n; i++) {
```

```
    a--;
```

```
}
```

d)  $2n^3 + 5$

```
for (int i = 0; i < n; i++) {
```

```
    for (int j = 0; j < n; j++) {
```

```
        for (int k = 0; k < n; k++) {
```

```
            a--; b--;
```

```
        }
```

```
    }
```

```
}
```

```
for (int i = 0; i < 5; i++) {
```

```
    a--;
```

```
}
```

e)  $9n^4 + 5n^2 + n/2$

```
for (int i = 0; i < n; i++) {
```

```
    for (int j = 0; j < n; j++) {
```

```
        for (int k = 0; k < n; k++) {
```

```
            for (int l = 0; l < n; l++) {
```

```
                a--; b--; c--; d--;
```

```
                e--; f--; g--; h--; i--;
```

```
            }
```

```
        }
```

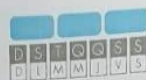
```
    }
```

```
}
```

10111

x: 0111

y: 01011



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

$f: \lg(n) + 5 \lg(n)$

```
for(int i=n; i>0; i/=2){
    a--;
}
```

```
for(int i=n; i>0; i/=2){
    a--; b--; c--; d--;
}
```

Exercício Resolvido 11:

1º - A operação relevante é a comparação entre elementos do array.

2º - Se houver  $n$  elementos,  $n-1$  vezes.

3º - Para o caso  $(\min > \text{array}[i])$ .

4º

☐

☐

☐

