

Exercício Resolvido (1): Resolva as Equações

- | | | |
|--------|---------|---------|
| a) 1 | b) 2 | c) 4 |
| d) 8 | e) 16 | f) 32 |
| g) 64 | h) 128 | i) 256 |
| j) 512 | k) 1024 | l) 2048 |

Exercício Resolvido (2): Resolva as Equações

- | | | |
|-------|-------|------|
| a) 11 | b) 10 | c) 9 |
| d) 8 | e) 7 | f) 6 |
| g) 5 | h) 4 | i) 3 |
| j) 2 | k) 1 | l) 0 |

Exercício Resolvido (3): Resolva as Equações

- | | | | | |
|----------------------|------------------|------------------|----------------------|------------------|
| a) $4,01 = 5$ | h) $\lg(17) = 5$ | k) $\lg(15) = 4$ | c) $4,99 = 5$ | f) $\lg(16) = 4$ |
| i) $\lg(17) = 4$ | l) $\lg(15) = 3$ | d) $4,99 = 4$ | g) $\lg(17) = 4,087$ | |
| j) $\lg(15) = 3,907$ | b) $4,01 = 4$ | e) $\lg(16) = 4$ | | |

Exercício Resolvido (6) = 3

Exercício Resolvido (7) = No melhor caso 3; No pior caso 5;

Exercício Resolvido (8) = Pior caso 7; Melhor caso 5;

Exercício Resolvido (9) = 4

Exercício Resolvido (10) = $n \cdot 2$

Exercício Resolvido (11) = $n - 3$

Exercício Resolvido (12) = 3

Exercício Resolvido (13) = 3

Exercício Resolvido (14) = 6

Exercício Resolvido (15) = $\lg(n) + 1$ vezes (no caso 6)

Exercício Resolvido (16)

- Faça um método que receba um número inteiro n e efetue o número de subtrações pedido em:

a) $3n + 2n^2$

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

//3n

//2n^2

b) $5n + 4n^3$

```
...  
i = 0;  
while (i < n){  
    i++;  
    a--; b--; c--; d--; e--;  
}  
for (i = 0; i < n; i++){  
    for (j = 0; j < n; j++){  
        for (k = 0; k < n; k++, a--, b--, c--, d--);  
    }  
}
```

//5n

4n^3

c) $\lg(n) + n$

```

...
i = 1;
while (i < n){ //log(n)
    i *= 2;
    a--;
}
for (i = 0; i < n; i++){ //n
    a--;
}

```

d) $2n^3 + 5$

```

...
for (i = 0; i < n; i++){ //2n^3
    for (j = 0; j < n; j++){
        for (k = 0; k < n; k++, a--, b--);
    }
}

a--; b--; c--; d--; e--; //5

```

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

```

...
for (i = 0; i < n; i++){ //2n^4
    for (j = 0; j < n; j++){
        a--; b--;
        for (k = 0; k < n; k++){ //2n^2
            for (l = 0; l < n; l++, c--; d--);
        }
    }
    if (i % 2 == 0) e--; //n%2
}

```

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

```
...  
i = 1;  
while (i < n){  
    i *= 2;  
    a--; b--; c--; d--; e--; f--;  
}
```

Exercício Resolvido (20):

Como em seu laço for, para o método de seleção, queremos que para antes da última casa do vetor, ou seja, além do -1 normal do for, que seria " $<n$ ", temos mais um -1 . E após isso, o laço swap afz 3 movimentações, ou seja $3(n-1)$.

Exercício Resolvido (23): melhor caso N , pior caso $2n$;

Exercício (1) = o número total de subtrações realizadas pelo código é 4.

Exercício (6) = $(n^2 - 7n)$;

Exercício (7) = $(n - 7)^2$;

Exercício (8) = $\log_2(n)$;

Exercício (9) = $\log_2(n + 1)$;

Exercício (10) = $\log_2(n)$;

Exercício (11) = $\log_2(n)$;