

TT 3 - Unidade 1a - Noções de complexidade

- **Exercícios iniciais**

- **Exercício 1**

- Resolva as equações abaixo:

a) $2^0 =$

d) $2^3 =$

g) $2^6 =$

j) $2^9 =$

b) $2^1 =$

e) $2^4 =$

h) $2^7 =$

k) $2^{10} =$

c) $2^2 =$

f) $2^5 =$

i) $2^8 =$

l) $2^{11} =$

Resposta:

a) 0	d) 8	g) 64	j) 512
b) 2	e) 16	h) 128	k) 1024
c) 4	f) 32	i) 256	l) 2048

- Exercício 2

• Resolva as equações abaixo:

$$\begin{array}{llll} \text{a) } \lg(2048) = & \text{d) } \lg(256) = & \text{g) } \lg(32) = & \text{j) } \lg(4) = \\ \text{b) } \lg(1024) = & \text{e) } \lg(128) = & \text{h) } \lg(16) = & \text{k) } \lg(2) = \\ \text{c) } \lg(512) = & \text{f) } \lg(64) = & \text{i) } \lg(8) = & \text{l) } \lg(1) = \end{array}$$

Nota: $\lg(n)$ é a mesma coisa que o logaritmo de n na base dois, ou seja, $\log_2(n)$

Algoritmos e Estruturas de Dados II (5)

Resposta:

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

- Exercício 3

• Resolva as equações abaixo:

$$\begin{array}{llll} \text{a) } \lceil 4,01 \rceil = & \text{d) } \lfloor 4,99 \rfloor = & \text{g) } \lg(17) = & \text{j) } \lg(15) = \\ \text{b) } \lfloor 4,01 \rfloor = & \text{e) } \lceil \lg(16) \rceil = & \text{h) } \lceil \lg(17) \rceil = & \text{k) } \lceil \lg(15) \rceil = \\ \text{c) } \lceil 4,99 \rceil = & \text{f) } \lfloor \lg(16) \rfloor = & \text{i) } \lfloor \lg(17) \rfloor = & \text{l) } \lfloor \lg(15) \rfloor = \end{array}$$

Algoritmos e Estruturas de Dados II (6)

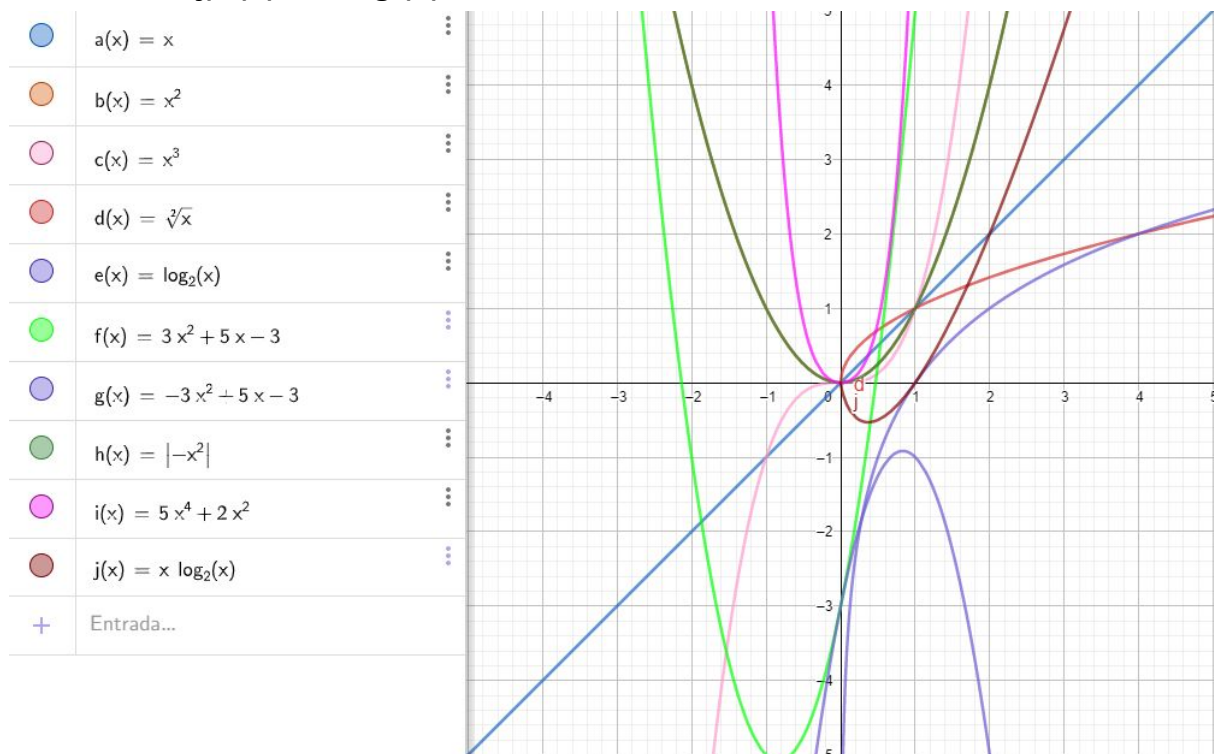
Resposta:

a) 5	d) 4	g)	j)
b) 4	e) 4	h)	k)
c) 5	f) 4	i)	l)

- Exercício 4

Plote um gráfico com todas as funções abaixo:

- a) $f(n) = n$
- b) $f(n) = n^2$
- c) $f(n) = n^3$
- d) $f(n) = \text{sqrt}(n)$
- e) $f(n) = \lg(n) = \log_2(n)$
- f) $f(n) = 3n^2 + 5n - 3$
- g) $f(n) = -3n^2 + 5n - 3$
- h) $f(n) = |-n^2|$
- i) $f(n) = 5n^4 + 2n^2$
- j) $f(n) = n * \lg(n)$



- **Contagem de operações:**

- **Exercício resolvido 1**

Calcule o número de subtrações que o código abaixo realiza:

```
a--;  
a -= 3;  
a = a - 2;
```

Resposta:

3 subtracoes.

- **Exercício resolvido 2**

Calcule o número de adições que o código abaixo realiza:

```
if (a + 5 < b + 3){  
    i++;  
    ++b;  
    a += 3;  
} else {  
    j++;  
}
```

Resposta:

Melhor caso : 3 vezes

Pior caso : 5 vezes

- **Exercício resolvido 3**

Calcule o número de adições que o código abaixo realiza:

```
if (a + 5 < b + 3 || c + 1 < d + 3){  
    i++;  
    ++b;  
    a += 3;  
} else {  
    j++;  
}
```

Resposta:

Melhor caso : 5 vezes

Pior caso : 7 vezes

- Exercício resolvido 4

Calcule o número de subtrações que o código abaixo realiza:

```
for (int i = 0; i < 4; i++){  
    a--;  
}
```

Resposta:

4 subtrações .

- Exercício resolvido 5

Calcule o número de subtrações que o código abaixo realiza:

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

Resposta:

2n subtrações.

- Exercício resolvido 6

Calcule o número de subtrações que o código abaixo realiza:

```
int i = 0, b = 10;  
  
while (i < 3){  
    i++;  
    b--;  
}
```

Resposta:

3 subtrações.

- Exercício resolvido 7

Calcule o número de subtrações que o código abaixo realiza:

```
for (int i = 3; i < n; i++){  
    a--;  
}
```

Resposta:

n - 3 subtrações.

- Exercício 5

Calcule o número de subtrações que o código abaixo realiza:

```
int i = 10;  
while (i >= 7){  
    i--;  
}
```

Resposta:

4 subtrações.

- Exercício 6

Calcule o número de subtrações que o código abaixo realiza:

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

Resposta:

8 subtrações.

- Exercício 7

Calcule o número de subtrações que o código abaixo realiza:

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

Resposta:

9 subtrações.

- Exercício 8

Calcule o número de subtrações que o código abaixo realiza:

```
int a = 10;
for (int i = 0; i < 3; i++){
    for (int j = 0; j < 2; j++){
        a--;
    }
}
```

Resposta:

6 subtrações.

- Exercício 9

Calcule o número de subtrações que o código abaixo realiza:

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

Resposta:

9 subtrações.

- Exercício 10

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$n * (n - 3)$ multiplicações .

- Exercício 11

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$(n - 7) * n$ multiplicações .

- Exercício 12

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n) + 1$ se n for potência de 2

$\lg(n) + 1$ (piso) se n não for potência de 2

- Exercício 13

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n + 4)$ multiplicações .

- Exercício 14

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$(n - 7)^2$ multiplicações .

- Exercício resolvido 9

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n) + 1$ se n for potência de 2

$\lg(n) + 1$ (piso) se n não for potência de 2

- Exercício 15

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n + 1) + 1$ se $n + 1$ for potência de 2

$\lg(n + 1) + 1$ (piso) se $n + 1$ não for potência de 2

- Exercício 16

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n)$, se $n + 1$ for potência de 2

$\lg(n)$, (piso) se $n + 1$ não for potência de 2

- Exercício 17

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n)$, se $n + 1$ for potência de 2

- Exercício 18

Calcule o número de multiplicações que o código abaixo realiza:

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

Resposta:

$\lg(n)$, se $n + 1$ for potência de 2

- Exercício resolvido 10

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

a) $3n + 2n^2$

```
static void a(int n) {
    int a = 0, b = 0, c = 0;

    for (int i = 0; i < n; i++)
        a--;b--;c--;

    int e = 0, f = 0;

    for (int j = 0; j < n; j++) {
        for (int k = 0; k < 2; k++)
            e--;f--;
    }
}
```

b) $5n + 4n^3$

```
static void b(int n) {
    int i = 0, j, k;
    int[] _5n = new int[5];
    while (i < n) {
        i++;
        for (j = 0; j < _5n.length; j++)
            _5n[j]--;
    }

    int[] _4n3 = new int[4];
    for(i = 0; i < n; i++)
        for(j = 0; j < n; j++)
            for(k = 0; k < _4n3.length; k++)
                _4n3[k]--;
}
```

```
}
```

c) $\lg(n) + n$

```
static void c(int n) {  
    int lgn = 0;  
    for(int i = n; i > 1; i/= 2)  
        lgn--;  
  
    for(int j = n; j > 0; j--){}  
}
```

d) $2n^3 + 5$

```
static void d(int n) {  
    int _2n3 = 0;  
    for(int i = 0; i < n; i++)  
        for(int j = 0; j < n; j++)  
            for(int k = 0; k < n; k++)  
                _2n3--;  
  
    int i = 5;  
    while(i > 0){  
        i--;  
    }  
}
```

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

```
static void e(int n) {  
    int _9n4 = 0;  
  
    for(int i = 0; i < Math.pow(n, 4); i++)  
        for(int j = 0; j < 9; j++)  
            _9n4--;  
  
    int _5n2 = 5 * (int)Math.pow(n, 2);  
    while(_5n2 > 0)  
        _5n2--;  
  
    int n2 = n/2;  
    while(n2 > 0)  
        n2--;  
}
```

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

```
static void f(int n) {  
    int lgn = 0;  
    for(int i = n; i > 1; i/= 2)  
        lgn--;  
  
    int _5lgn = 0;  
    for(int i = 0; i < 5; i++)  
        for(int j = n; j > 1; j/= 2)  
            _5lgn--;  
  
}
```

- Exercício resolvido 11

Encontre o menor valor em um array de inteiros:

```
int min = array[0];  
  
for (int i = 1; i < n; i++){  
    if (min > array[i]){  
        min = array[i];  
    }  
}
```

1) **Qual é operação relevante?**

r. Comparação entre min e array[i]

2) **Quantas vezes ela será executada?**

r. Ela será executada $n - 1$ vezes

3) **O nosso $T(n) = n - 1$ é para qual dos três casos?**

r. Será para a comparação

- Exercício 19

Qual é a diferença entre as notações O , Ω e Θ ?

- O = Significa o pior caso
- Ω = Significa o melhor caso
- Θ = Significa o caso medio