

Noções de Complexidade

(GABARITO)

Unidade I: Análise de Algoritmos

Exercício (1)

- Calcule o **número de subtrações** que o código abaixo realiza:

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

Exercício (1)

- Calcule o **número de subtrações** que o código abaixo realiza:

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

Exercício (2)

- Calcule o **número de subtrações** que o código abaixo realiza:

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

Exercício (2)

- Calcule o **número de subtrações** que o código abaixo realiza:

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

Exercício (3)

- 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--;  
    }  
}
```

Exercício (3)

- 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--;  
    }  
} // Oito subtrações
```

Exercício (4)

- Calcule o **número de subtrações** que o código abaixo realiza:

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


Exercício (4)

- Calcule o **número de subtrações** que o código abaixo realiza:

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

Exercício (5)

- 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;  
    }  
}
```

Exercício (5)

- 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;  
    }  
} //n(n-3) multiplicações
```

Exercício (6)

- 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;  
    }  
}
```

Exercício (6)

- 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;  
    }  
} // (n-7)n multiplicações
```

Exercício (7)

- 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;  
    }  
}
```

Exercício (7)

- 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;  
    }  
} // (n-7)2 multiplicações
```

Exercício (8)

- Calcule o **número de multiplicações** que o código abaixo realiza:

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


Exercício (8)

- Calcule o **número de multiplicações** que o código abaixo realiza:

```
...  
for (int i = n; i > 1; i /= 2){  
    a *= 2;  
} // piso(lg(n)) + 1 multiplicações
```

Exercício (9)

- Calcule o **número de multiplicações** que o código abaixo realiza:

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

Exercício (9)

- Calcule o **número de multiplicações** que o código abaixo realiza:

```
...  
for (int i = n + 1; i > 0; i /= 2)  
    a *= 2;  
} // piso(lg(n+1)) + 1 multiplicações
```

Exercício (10)

- Calcule o **número de multiplicações** que o código abaixo realiza:

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

Exercício (10)

- Calcule o **número de multiplicações** que o código abaixo realiza:

```
...  
for (int i = 1; i < n; i *= 2)  
    a *= 2;  
}//teto(lg(n)) multiplicações
```

Exercício (11)

- Calcule o **número de multiplicações** que o código abaixo realiza:

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

Exercício (11)

- Calcule o **número de multiplicações** que o código abaixo realiza:

```
...  
for (int i = 1; i <= n; i *= 2)  
    a *= 2;  
} // teto(lg(n)) + 1 multiplicações
```

Exercício (12)

- Calcule o **número de multiplicações** que o código abaixo realiza:

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


Exercício (12)

- Calcule o **número de multiplicações** que o código abaixo realiza:

```
...  
for (int i = n+4; i > 0; i >>= 1){  
    a *= 2;  
} // piso(lg(n+4)) + 1 multiplicações
```