



## Ejercicios

*Este documento se hizo con propósitos educativos, mucho del conocimiento e información del mismo se adjudica al Ing. Luis Humberto González Guerra.*

Ozym4ndias | Roberto A. Pérez

=====

### **Orden Lineal y logaritmica**

- Nos fijamos que esté siempre en términos de  $n$

### **Orden constante**

- No esta en términos de  $n$

=====

```

// ----ORDEN LINEAL----
// =====
for (int i = 0; i < n; i++){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = 0; i > 0; i--){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = 1; i < n; i += 5){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = n; i > 0; i -= 5){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = 1; i < n/5; i++){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = n; i > n/3; i--){ // Es de orden lineal
    // Instrucción;
}
// =====
for (int i = 4; i < n/5; i += 3){ // Es de orden lineal
    // Instrucción;
}
// =====

// ----ORDEN CONSTANTE----
// =====
for (int i = 4; i < 80; i += 3){ // Es de orden constante
    // Instrucción;
}
// =====

```

=====

```
// ----ORDEN CONSTANTE----
// =====
for (int i = 4; i < 80; i += 3){ // Es de orden constante
    // Instrucción;
}
// =====
```

=====

```
// ----ORDEN LOGARITMICO----
// =====
for (int i = 1; i <= n; i *= 2){ // Es de orden logaritmico
    // Instrucción;
}
// =====
for (i = n; i > 0; i /= 2){ // Es de orden logaritmico
    // Instrucción;
}
```

=====

```

// =====
// Anidado vs secuencia
// =====

// ----ANIDADO----
// Anidado se multiplica
for (int i = 1; i <= n; i++){ // Orden lineal  $O(n)$ 
    for (int j = 1; j < n; j++){ // Orden lineal  $O(n)$ 
        // Instrucción;
    }
}

//  $O(n) * O(n) \rightarrow O(n^2)$ 
// =====
for (int i = n; i > 5; i /= 2){ // Orden logaritmico  $O(\log n)$ 
    for (int j = 1; j < n/5; j += 3){ // Orden lineal  $O(n)$ 
        // Instrucción;
    }
}

//  $O(\log n) * O(n) \rightarrow O(n \log n)$ 
// =====
for (int i = 2; i < n/2; i *= 2){ // Orden logaritmico  $O(\log_2 n)$ 
    for (int j = n; j > 0; j /= 2){ // Orden logaritmico  $O(\log_2 n)$ 
        // Instrucción;
    }
}

//  $O(\log_2 n) * O(\log_2 n) \rightarrow O(\log_2 n)^2$ 
// =====
for (int i = 2; i < n/2; i *= 2){ // Orden logaritmico  $O(\log n)$ 
    for (int j = n; j > 0; j /= 3){ // Orden logaritmico  $O(\log n)$ 
        // Instrucción;
    }
}

//  $O(\log_2 n) * O(\log_3 n) \rightarrow O(\log_2 n * \log_3 n)$ 
// =====

// ----Secuencial----
// Secuencial se elige al peor (mayor)
for (int i = 1; i <= n; i++){ // Orden lineal  $O(n)$ 
    // Instrucción;
}

for (int i = n; i > 0; i /= 2){ // Orden logaritmico  $O(\log n)$ 
    // Instrucción;
}

// Orden lineal es peor que orden logaritmico, por lo tanto se elige  $O(n)$ 
// =====

```

=====

Ejercicio hecho a mano por el Ing. Luis Humberto González

$O(n^2)$

$O(n^2)$

$O(n)$

```
for (int i = 1; i < n/2; i++) {  
    for (int j = n; j > 0; j--) {  
        Instr:  $O(1)$   
        if (n % 2 == 0) {  
            for (int k = n; k > 0; k--) {  
                Instr:  $O(1)$   
                for (int l = 1; l < 80; l++) {  
                    Instr:  $O(1)$   
                }  
            }  
        }  
        else {  
            for (int z = n; z > 0; z--) {  
                Instr:  $O(1)$   
            }  
        }  
    }  
}
```

Pause