

1. Complejidad

1.1. Análisis de frecuencia

1. 1. a) Caso base bucles for

```
for(int i = 0; i < n; i++) {  
    System.out.println(i);  
}
```

$n + 1$ n Dentro del bucle se resta -1 Se ejecuta n veces

Variaciones

```
for(int i = 0; i <= n; i++) {  
    System.out.println(i);  
}
```

$n + 2$ Debido al \leq , se suma $+1$
 $n + 1$

```
for(int i = 1; i < n; i++) {  
    System.out.println(i);  
}
```

n Debido a que empieza en 1, se resta -1
 $n - 1$

```
for(int i = 5; i < n; i++) {  
    System.out.println(i);  
}
```

$n - 4$ Debido a que empieza en 5, se resta -5
 $n - 5$

```
for(int i = 0; i < n + 2; i++) {  
    System.out.println(i);  
}
```

$n + 3$ Debido al que itera hasta $n + 2$, se suma $+2$
 $n + 2$

1. 1. b) Caso base bucle while

```
while (n > 0) {  
    System.out.println(n);  
    n--;  
}
```

$n + 1$ n Dentro del bucle se resta -1 Se ejecuta n veces
 n

Variaciones

```
while (n >= 0) {  
    System.out.println(n);  
    n = n - 2;  
}
```

$n + 2$ Debido al \geq , se suma $+1$
 $n + 1$
 $n + 1$

```
while (n > 2) {  
    System.out.println(n);  
    n--;  
}
```

$n - 1$ Debido al $n > 2$, se resta -2
 $n - 2$
 $n - 2$

```
while (n > 0) {  
    System.out.println(n);  
    n = n - 2;  
}
```

$n - 1$ $n - 2$ Debido al $n = n - 2$, se resta -2
 $n - 2$

```
while (n > 0) {  
    System.out.println(n);  
    n = n * 2;  
}
```

$\log n + 1$ $\log n$ Debido al $n = n * 2$, se aplica $\log n$
 $\log n$

```
while (n > 0) {
    System.out.println(n);
    n = n / 2;
}
```

$\log n + 1$
 $\log n$
 $\log n$

Debido al $n = n / 2$, se aplica $\log n$

1.2. Algoritmos recursivos

1. 2. a) Método iterativo

$$T(n) = \begin{cases} 4 & \text{si } n = 2 \\ T(n - 5) + 11 & \text{si } n > 2 \end{cases}$$

Se debe repetir la función hasta encontrar un patrón con k

$$\begin{aligned} T(n) &= T(n - 5) + 11 & T(n - 5) &= T(n - 10) + 11 \\ T(n) &= T(n - 10) + 22 & T(n - 10) &= T(n - 15) + 11 \\ T(n) &= T(n - 15) + 33 \end{aligned}$$

El patrón es:

$$T(n) = T(n - 5k) + 11k$$

Si igualamos $T(n - 5k) = T(2)$:

$$\begin{aligned} n - 5k &= 2 \\ -5k &= 2 - n \\ k &= -\frac{2 - n}{5} \end{aligned}$$

Ahora

$$\begin{aligned} T(n) &= T(2) + 11k \\ T(n) &= 4 + 11\left(-\frac{2 - n}{5}\right) \\ T(n) &= 4 - \frac{22 - 11n}{5} \\ T(n) &= \frac{20 - (22 - 11n)}{5} \\ T(n) &= \frac{-2 + 11n}{5} \end{aligned}$$

Esto da un orden de complejidad $O(n)$ lineal

1. 2. b) Teorema máster

$$\begin{aligned} T(n) &= aT\left(\frac{n}{b}\right) + cn^k \\ T(n) &\in \begin{cases} O(n^{\log_b a}) & \text{si } a > b^k \\ O(n^k \log n) & \text{si } a = b^k \\ O(n^k) & \text{si } a < b^k \end{cases} \end{aligned}$$

2. Segment tree

