

# Introducción a la Computación (Matemática)

Primer Cuatrimestre de 2019

Clase Práctica: Complejidad Algorítmica

# Ejemplo 1.1

```
 $RV \leftarrow 0$   
 $i \leftarrow 0$   
while ( $i < |A|$ ) {  
     $j \leftarrow 0$   
     $sumaAnteriores \leftarrow 0$   
    while ( $j < i$ ) {  
         $sumaAnteriores \leftarrow sumaAnteriores + A[j]$   
         $j \leftarrow j + 1$   
    }  
    if ( $sumaAnteriores = A[i]$ ) {  
         $RV \leftarrow RV + 1$   
    }  
     $i \leftarrow i + 1$   
}
```

# Ejemplo 1.1

```
RV  $\leftarrow$  0      (1)
i  $\leftarrow$  0     (1)
while (i < |A|) { (3)
    j  $\leftarrow$  0  (1)
    sumaAnteriores  $\leftarrow$  0 (1)
    while (j < i) { (3)
        sumaAnteriores  $\leftarrow$  sumaAnteriores + A[j] (5)
        j  $\leftarrow$  j + 1 (3)
    }
    if (sumaAnteriores = A[i]) { (4)
        RV  $\leftarrow$  RV + 1 (3)
    }
    i  $\leftarrow$  i + 1 (3)
}
```

# Ejemplo 1.1

```
RV ← 0      (1)
i ← 0      (1)
while (i < |A|) {      (3)
    j ← 0      (1)
    sumaAnteriores ← 0      (1)
    while (j < i) {      (3)
        sumaAnteriores ← sumaAnteriores + A[j]      (5)
        j ← j + 1      (3)
    }
    if (sumaAnteriores = A[i]) {      (4)
        RV ← RV + 1      (3)
    }
    i ← i + 1      (3)
}
```

$$\begin{aligned} T(|A|) &= 1 + 1 + 3 + \sum_{0 \leq i < |A|} (1 + 1 + 3 + \sum_{0 \leq j < i} (5 + 3 + 3) + 4 + 3 + 3 + 3) \\ &= 5 + \sum_{0 \leq i < |A|} (18 + \sum_{0 \leq j < i} 11) = 5 + 18|A| + \sum_{0 \leq i < |A|} i \cdot 11 \\ &= 5 + 18|A| + \frac{11}{2} (|A| - 1)|A| = 5 + \frac{25}{2}|A| + \frac{11}{2}|A|^2 \end{aligned}$$

## Ejemplo 1.2

$RV \leftarrow 0$

$i \leftarrow 0$

$sumaAnteriores \leftarrow 0$

while ( $i < |A|$ ) {

    if ( $sumaAnteriores = A[i]$ ) {

$RV \leftarrow RV + 1$

    }

$sumaAnteriores \leftarrow sumaAnteriores + A[i]$

$i \leftarrow i + 1$

}

## Ejemplo 1.2

```
 $RV \leftarrow 0$  (1)  
 $i \leftarrow 0$  (1)  
 $sumaAnteriores \leftarrow 0$  (1)  
while ( $i < |A|$ ) { (3)  
    if ( $sumaAnteriores = A[i]$ ) { (4)  
         $RV \leftarrow RV + 1$  (3)  
    }  
     $sumaAnteriores \leftarrow sumaAnteriores + A[i]$  (5)  
     $i \leftarrow i + 1$  (3)  
}
```

## Ejemplo 1.2

```
 $RV \leftarrow 0$  (1)  
 $i \leftarrow 0$  (1)  
 $sumaAnteriores \leftarrow 0$  (1)  
while ( $i < |A|$ ) { (3)  
    if ( $sumaAnteriores = A[i]$ ) { (4)  
         $RV \leftarrow RV + 1$  (3)  
    }  
     $sumaAnteriores \leftarrow sumaAnteriores + A[i]$  (5)  
     $i \leftarrow i + 1$  (3)  
}
```

$$\begin{aligned} T(|A|) &= 1 + 1 + 1 + 3 + \sum_{0 \leq i < |A|} (4 + 3 + 5 + 3 + 3) \\ &= 6 + \sum_{0 \leq i < |A|} 18 = 6 + 18|A| \end{aligned}$$

# Orden del tiempo de ejecución

**Definición:** Decimos que  $T(n) \in O(f(n))$  si existen constantes enteras positivas  $c$  y  $n_0$  tales que para  $n \geq n_0$ ,  $T(n) \leq c \cdot f(n)$ .



# Orden del tiempo de ejecución

**Definición:** Decimos que  $T(n) \in O(f(n))$  si existen constantes enteras positivas  $c$  y  $n_0$  tales que para  $n \geq n_0$ ,  $T(n) \leq c \cdot f(n)$ .

**Ejemplo 1.1:**  $T(|A|) = 5 + \frac{25}{2} |A| + \frac{11}{2} |A|^2 \in O(|A|^2)$   
(orden cuadrático)

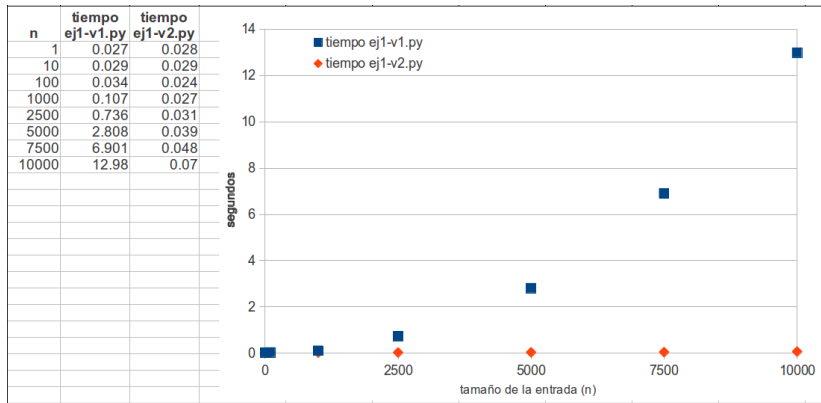
# Orden del tiempo de ejecución

**Definición:** Decimos que  $T(n) \in O(f(n))$  si existen constantes enteras positivas  $c$  y  $n_0$  tales que para  $n \geq n_0$ ,  $T(n) \leq c \cdot f(n)$ .

**Ejemplo 1.1:**  $T(|A|) = 5 + \frac{25}{2} |A| + \frac{11}{2} |A|^2 \in O(|A|^2)$   
(orden cuadrático)

**Ejemplo 1.2:**  $T(|A|) = 6 + 18 |A| \in O(|A|)$  (orden lineal)

# Tiempo de ejecución



Comparación de los tiempos de ejecución de implementaciones en Python de los algoritmos de los ejemplos 1.1 (*ej1-v1.py*) y 1.2 (*ej1-v2.py*).