

Análisis del problema de ordenamiento de secuencias

Leonardo Flórez-Valencia

2014-2019

1. Descripción y formalización del problema

El problema, informalmente, se define como: Ordenar una “lista” / “arreglo” / “vector” / “conjunto” / “montón” de números.

Formalmente, se dice que: Dada una secuencia S de elementos $a_i \in \mathbb{T}$, donde se define la relación de orden parcial \leq , producir una nueva secuencia S' donde los elementos contiguos cumplan la relación de orden parcial \leq . Ahora, la definición del contrato sería:

- **Entradas:** Una secuencia S de n números: $S = \langle a_1, a_2, \dots, a_n \rangle$ donde $a_i \in \mathbb{T}$ y en \mathbb{T} está definida la relación de orden parcial \leq .
- **Salidas:** Una permutación $S' = \langle a'_1, a'_2, \dots, a'_n \rangle \mid a'_1 \leq a'_2 \leq \dots \leq a'_n \wedge a'_i \in S \forall i$

2. Ordenamiento burbuja

Algorithm 1 Algoritmo de ordenamiento por burbuja.

```
1: procedure BUBBLESORT( $S$ )
2:   for  $i \leftarrow 1$  to  $|S|$  do
3:     for  $j \leftarrow 1$  to  $|S| - i$  do
4:       if  $S[j+1] < S[j]$  then
5:          $aux \leftarrow S[j]$ 
6:          $S[j] \leftarrow S[j+1]$ 
7:          $S[j+1] \leftarrow aux$ 
8:       end if
9:     end for
10:  end for
11: end procedure
```

3. Ordenamiento por inserción

Algorithm 2 Algoritmo de ordenamiento por inserción.

```
1: procedure INSERTIONSORT( $S$ )
2:   for  $j \leftarrow 2$  to  $|S|$  do
3:      $k \leftarrow S[j]$ 
4:      $i \leftarrow j - 1$ 
5:     while  $0 < i \wedge k < S[i]$  do
6:        $S[i + 1] \leftarrow S[i]$ 
7:        $i \leftarrow i - 1$ 
8:     end while
9:      $S[i + 1] \leftarrow k$ 
10:  end for
11: end procedure
```
