

Algoritmos

- Ordenamiento

- ❑ Algoritmo Burbuja

- ❑ Su nombre proviene del hecho de que los elementos mas “livianos” (pequeños) salgan a flote, mientras que los mas pesados “grandes” se hundan.

Algoritmos

- Ordenamiento

- Algoritmo Burbuja

- Ejemplo 1:

- Arreglo A

12	5	34	65	10
----	---	----	----	----

- Si el dato en la posición inferior es más pequeño se cambian los valores de posición y se continúa con el siguiente número. Luego que termina de dar el primer recorrido al arreglo, comienza de nuevo desde el principio, pero el elemento más pesado ya quedó al fondo, por lo tanto, ahora se comparan sólo los primeros N-1 elementos

Algoritmos

○ Ordenamiento

□ Algoritmo Burbuja

□ Ejemplo 1:

	12	5	34	65	10
$12 < 5$	5	12	34	65	10
$12 < 34$	5	12	34	65	10
$34 < 65$	5	12	34	65	10
$65 < 10$	5	12	34	10	65

El número **65** ya quedo en su posición final, al ser el elemento mas grande

Se repite para los N-1 restantes

Algoritmos

- Ordenamiento

- Algoritmo Burbuja

- Ejemplo 1:

	5	12	10	34	65
5 < 12	5	12	34	10	65
12 < 34	5	12	34	10	65
34 < 10	5	12	10	34	65
34 < 65	5	12	10	34	65

Se continúa hasta obtener el arreglo ordenado

Algoritmos

○ Ordenamiento

□ Algoritmo Burbuja

□ Ejemplo 1:

A

15	8	8	8	8	8	4	4	4	4	4	4	3
8	15	4	4	4	4	8	8	8	8	8	3	4
4	4	15	15	15	15	15	15	3	3	3	8	8
50	50	50	50	3	3	3	3	15	15	15	15	15
3	3	3	3	50	50	50	50	50	50	50	50	50

Algoritmos

○ Ordenamiento

□ Algoritmo Burbuja

```
...
1. HACER  $i \leftarrow N$ 
2. REPETIR CON  $i$  DESDE  $N$  HASTA 2
  2.1. HACER  $j \leftarrow 1$ 
  2.2. REPETIR CON  $j$  DESDE 1 HASTA  $i - 1$ 
    2.2.1 SI  $A[j] > A[j+1]$  ENTONCES
      2.2.2.1 HACER  $temp \leftarrow A[j]$ ;
      2.2.2.2 HACER  $A[j] \leftarrow A[j+1]$ ;
      2.2.2.3 HACER  $A[j+1] \leftarrow temp$ ;
    2.2.2. {Fin del condicional del paso 2.2.1.}
  2.2.3. HACER  $j \leftarrow j + 1$ 
  2.3. {Fin del ciclo del paso 2.2.}
  2.4. HACER  $i \leftarrow i - 1$ 
3. {Fin del ciclo del paso 2}
...
```