Métodos de Ordenamiento





Métodos de Ordenamiento

Métodos Iterativos:

- Burbuja
 - Simple
 - Mejorada
- Inserción
- Selección



Métodos Iterativos

Burbuja Simple:

Este método es uno de los más simples, es tan fácil como comparar todos los elementos de una lista (arreglo) contra todos, si se cumple que uno es mayor o menor a otro, entonces los intercambia de posición.



Ejemplo:

ESTRATEGIA

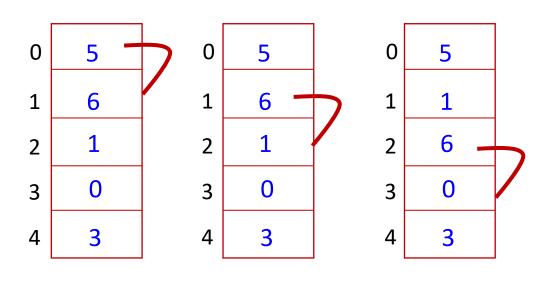
Comparamos todos contra todos, aunque estemos generando comparaciones extras.



Ejemplo:

Paso 1

para i = 0



0	5
1	1
2	0
3	3
4	6

j=0

j=1

j=2

j=3

3

0

3

4

5

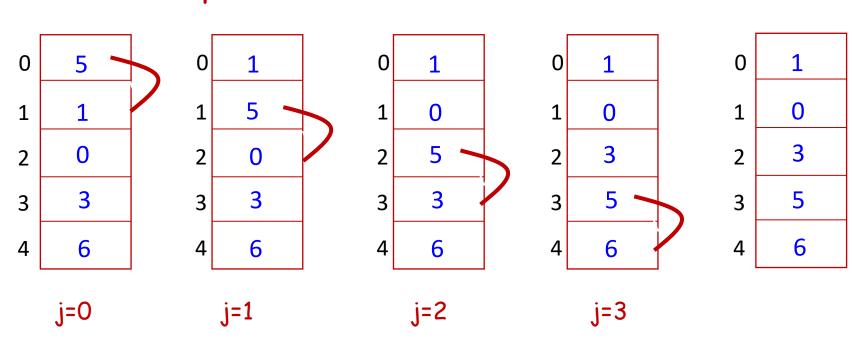
0



Ejemplo:

Paso 2

para i = 1

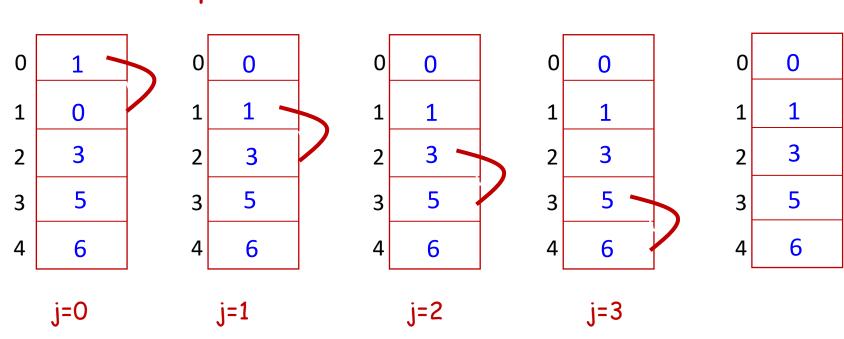




Ejemplo:

Paso 3

para i = 2





```
public static void simpleBubbleSort(int[] A, int n) {
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n - 1; j++) {
        if(A[j] > A[j + 1]){
             int aux = A[j];
            A[j] = A[j + 1];
            A[j + 1] = aux;
```



Método de Selección

Este algoritmo consiste en lo siguiente:

- Buscar el elemento más pequeño de la lista.
- Intercambiarlo con el elemento ubicado en la primera posición de la lista.
- Buscar el segundo elemento más pequeño de la lista.
- Intercambiarlo con el elemento que ocupa la segunda posición en la lista.
- Repetir este proceso hasta que toda la lista se encuentre ordenada.

Método de Selección

Por ejemplo:

Lista a ordenar: <u>4</u> - 3 - 5 - 2 - 1

intercambiamos 1 con el 4 y la lista queda así:

Lo intercambiamos 2 con el elemento en la segunda posición, es decir el 3. La lista queda así:

intercambiamos 3 con el 5:

intercambiamos 4 con el 5:



Método de Selección

```
public static void selectionSort(int[] A, int n) {
for (int i = 0; i < n; i++) {
    int p = i;
    for (int j = i + 1; j < n; j++) {
        if(A[j] < A[p]){
            p = j;
    if(p != i){
        int temp = A[p];
        A[p] = A[i];
        A[i] = temp;
```



Método de Inserción

Consideremos que en un momento dado, el segmento izquierdo del arreglo está ordenado y el segmento derecho no.

Podemos tomar un elemento del segmento derecho e *insertarlo* en su lugar correcto en el lado izquierdo.

Parte ordenada Parte desordenada



Métodos de Inserción

- > Se considera la lista partida en dos sublistas, una ordenada y la otra para ordenar
- > En el primer paso, la lista ordenada contiene un elemento y la lista por ordenar, n-1,
- > Se van extrayendo uno a uno los elementos de la lista desordenada y se colocan en la sublista ordenada, mediante comparaciones sucesivas, hasta encontrar la posición correcta.
- > El proceso continúa hasta que la lista desordenada queda vacía.



Métodos de Inserción

Ejemplo:

```
8 12 4 9 3
```

8 12 4 9 3

4 8 12 9 3

4 8 9 12 3

3 4 8 9 12



Métodos Iterativos

```
public static void insertionSort(int[] A, int n){
for (int i = 0; i < n; i++) {
    int current = A[i];
    int position = i;
    while(position > 0 && A[position - 1] > current){
        A[position] = A[position - 1];
        position--;
    A[posicion] = current;
```