

Algoritmos de Ordenación

1. Introducción

1.1 Algoritmos de Ordenación En computación y matemáticas un [algoritmo de ordenamiento](#) es un algoritmo que pone elementos de una lista o un vector en una secuencia dada por una relación de orden, es decir, el resultado de salida ha de ser una permutación o reordenamiento de la entrada que satisfaga la relación de orden dada. Las relaciones de orden más usadas son el orden numérico y el orden lexicográfico (Wikipedia). En esta primera parte mostraremos algunos algoritmos básicos de ordenación, codificados en Python.

2. Algoritmos

2.1 Ordenamiento de burbuja (Bubble Sort)

2.2 Ordenamiento por inserción (Insertion Sort)

2.3 Ordenamiento por mezcla (Merge Sort)

2.4 Ordenamiento rápido (QuickSort)

2.5 Ordenamiento por selección (Selection Sort)

2.6 Ordenamiento shell (Shell Sort)

3. Explicaciones

3.1 Ordenamiento de burbuja (Bubble Sort) El método de la burbuja es uno de los mas simples algoritmos de ordenación, es como comparar todos los elementos de una lista contra todos, si se cumple que uno es mayor o menor a otro, entonces se intercambia de posición.

```

·5·2·7·3·1
·2·5·7·3·1
·2·5·3·7·1
·2·5·3·1·7
·2·3·5·1·7
·2·3·1·5·7
·2·1·3·5·7
·1·2·3·5·7
    
```

Figure 1: Ordenamiento de burbuja.

3.2 Ordenamiento por inserción (Insertion Sort) Este método tiene la semejanza con la forma de clasificar las cartas de una baraja, insertando cada carta en el lugar adecuado. El algoritmo ordena los dos primeros elementos de la lista, a continuación el tercer elemento se inserta en la posición que corresponda, el cuarto se inserta en la lista de tres elementos, y así sucesivamente. Este proceso continua hasta que la lista este totalmente ordenada.

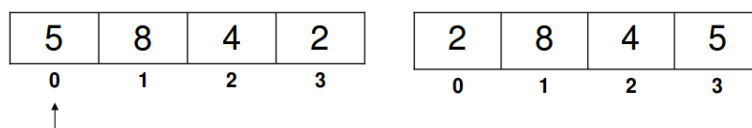
3	1	4	1	2	9	5	6	5	3
1	3	4	1	2	9	5	6	5	3
1	3	4	1	2	9	5	6	5	3
1	1	3	4	2	9	5	6	5	3
1	1	2	3	4	9	5	6	5	3
1	1	2	3	4	9	5	6	5	3
1	1	2	3	4	5	9	6	5	3
1	1	2	3	4	5	6	9	5	3
1	1	2	3	4	5	6	5	9	3
1	1	2	3	3	4	5	5	6	9

Figure 2: Ordenamiento por inserción.

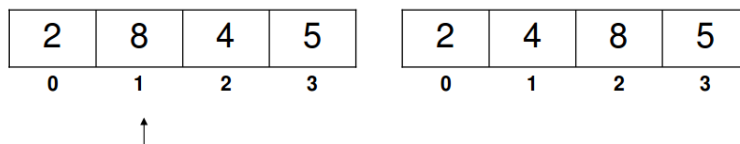
3.3 Ordenamiento por selección (Selection Sort) (Wikipedia) Es un algoritmo de ordenación que funciona de la siguiente manera:

- Busca el mínimo elemento entre una posición i y el final de la lista.
- Intercambiar el mínimo con el elemento de la posición i .

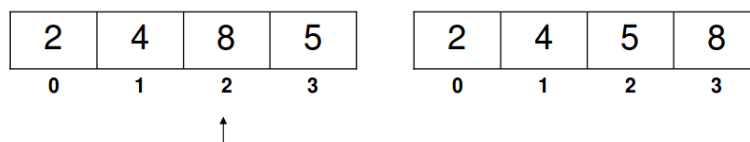
Por ejemplo, se recorre una lista, se selecciona el elemento menor y se intercambia este elemento con el de la primera posición.



Se hace lo mismo, pero ahora se busca desde la segunda posición hasta el final el menor. Se intercambia éste menor con lo que está en la segunda posición.



Se repite para las siguientes posiciones, hasta la posición $n - 1$, si es que la lista tiene longitud n .



3.4 Ordenamiento por mezcla (Merge Sort) El algoritmo de ordenamiento por mezcla (Merge Sort) es un algoritmo de ordenamiento basado en la **técnica divide y vencerás**, desarrollado por John Von Neumann.

(Wikipedia) El ordenamiento por mezcla funciona de la siguiente manera:

- Si la longitud de la lista es 0 ó 1, entonces ya está ordenada. En otro caso:
- Dividir la lista desordenada en dos sublistas de aproximadamente la mitad del tamaño.
- Ordenar cada sublista recursivamente aplicando el ordenamiento por mezcla.
- Mezclar las dos sublistas en una sola lista ordenada.

Si se piensa en este algoritmo recursivamente, podemos imaginar que dividirá la lista hasta tener un elemento en cada lista, luego lo compara con el que está a su lado y según corresponda, lo sitúa donde corresponde. En la figura se ve como funciona.

3.5 Ordenamiento rápido (QuickSort) (Wikipedia) El ordenamiento rápido (quicksort) es un algoritmo basado en la técnica de divide y vencerás, que permite, en promedio, ordenar n elementos en un tiempo proporcional a $n \log n$.

El algoritmo trabaja de la siguiente forma:

- Elegimos un elemento de la lista de elementos a ordenar, al que llamaremos pivote.
- Acomodamos los demás elementos de la lista a cada lado del pivote, de manera que a un lado queden todos los menores que él, y al otro los mayores. Los elementos iguales al pivote pueden ser colocados tanto a su derecha como a su izquierda, dependiendo de la implementación deseada. En este momento, el pivote ocupa exactamente el lugar que le corresponder en la lista ordenada.

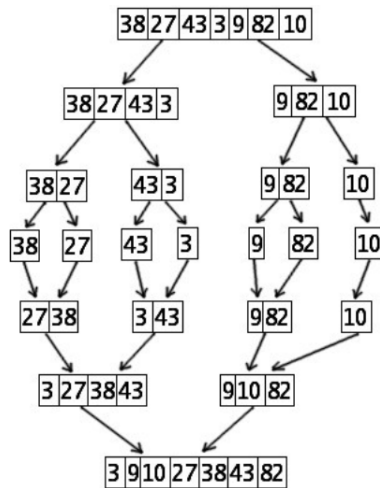


Figure 3: Ordenamiento por mezcla.

- La lista queda separada en dos sublistas, una formada por los elementos a la izquierda del pivote, y otra por los elementos a su derecha.
- Repetimos este proceso de forma recursiva para cada sublista mientras éstas contengan más de un elemento. Una vez terminado este proceso todos los elementos estarán ordenados.

La eficiencia del algoritmo depende de la posición en la que termine el pivote elegido. La siguiente figura muestra de mejor manera lo mencionado, usando como pivote el primer elemento.

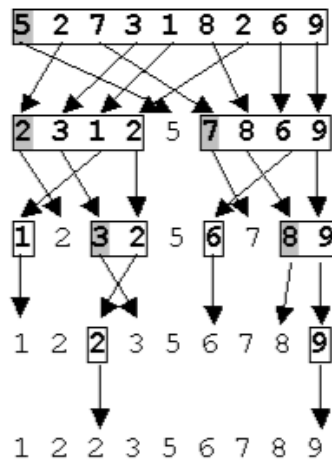


Figure 4: Ordenamiento rápido (quicksort).

3.6 Ordenamiento shell (Shell Sort) Es un algoritmo de ordenación muy ingenioso, basado en comparaciones e intercambios. (Ecured) el algoritmo Shell es una mejora de la ordenación por inserción, donde se van comparando elementos distantes, al tiempo que se los intercambian si corresponde. A medida que se aumentan los pasos, el tamaño de los saltos disminuye; por esto mismo, es útil tanto como si los datos desordenados se encuentran cercanos, o lejanos.

Es bastante adecuado para ordenar listas de tamaño moderado, debido a que su velocidad es aceptable y su codificación es sencilla.