

# Algoritmo De Ordenamiento Radix (Radix sort)

Es un algoritmo de ordenamiento que ordena enteros procesando sus dígitos de forma individual. Como los enteros pueden representar cadenas de caracteres (por ejemplo, nombres o fechas) y, especialmente, números en punto flotante especialmente formateados.

Este ordenamiento se basa en los valores de los dígitos reales en las representaciones de posiciones de los números que se ordenan.

Por ejemplo, el número 235 se escribe 2 en la posición de centenas, un 3 en la posición de decenas y un 5 en la posición de unidades.

Reglas para ordenar.

- Empezar en el dígito más significativo y avanzar por los dígitos menos significativos mientras coinciden los dígitos correspondientes en los dos números.
- El número con el dígito más grande en la primera posición en la cual los dígitos de los dos números no coinciden es el mayor de los dos (por supuesto sí coinciden todos los dígitos de ambos números, son iguales).

Este mismo principio se toma para Radix Sort, para visualizar esto mejor tenemos el siguiente ejemplo. En el ejemplo anterior se ordenó de izquierda a derecha. Ahora vamos a ordenar de derecha a izquierda.

## Pseudocódigo

```
def CreaLista(k):  
    L=[]  
    for i in range(k+1):  
        L.append(0)  
    return L  
  
def CountingSort(A,k):  
    C=CreaLista(k)  
    B=CreaLista(len(A)-1)  
    for j in range(1,len(A)):  
        C[A[j]]=C[A[j]]+1  
    for i in range(1,k+1):  
        C[i]=C[i]+C[i-1]  
    for j in range(len(A)-1,0,-1):  
        B[C[A[j]]]=A[j]  
        C[A[j]]=C[A[j]]-1  
    return B
```