**Celal Bayar Üniversitesi**

**Yazılım Mühendisliği**

**Algoritma Analizi ve Tasarımı Projesi**

**Ad: Muhammet Emin**

**Soyad: Çakır**

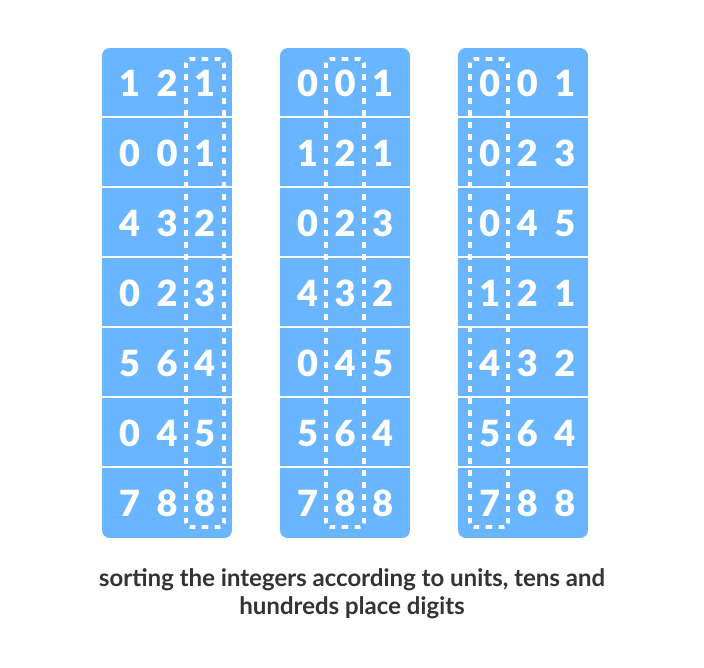
**Numara: 192802046**

**Radix Sıralama Algoritması**

Radix sıralama, önce aynı basamak **değerinin** tek tek basamaklarını gruplayarak öğeleri sıralayan [bir sıralama algoritmasıdır](https://www.programiz.com/dsa/sorting-algorithm) . Ardından elemanları artan/azalan sırasına göre sıralayın.

Diyelim ki 8 elemanlı bir dizimiz var. İlk olarak, öğeleri birim yerin değerine göre sıralayacağız. Ardından, onuncu yerin değerine göre öğeleri sıralayacağız. Bu işlem son önemli yere kadar devam eder.

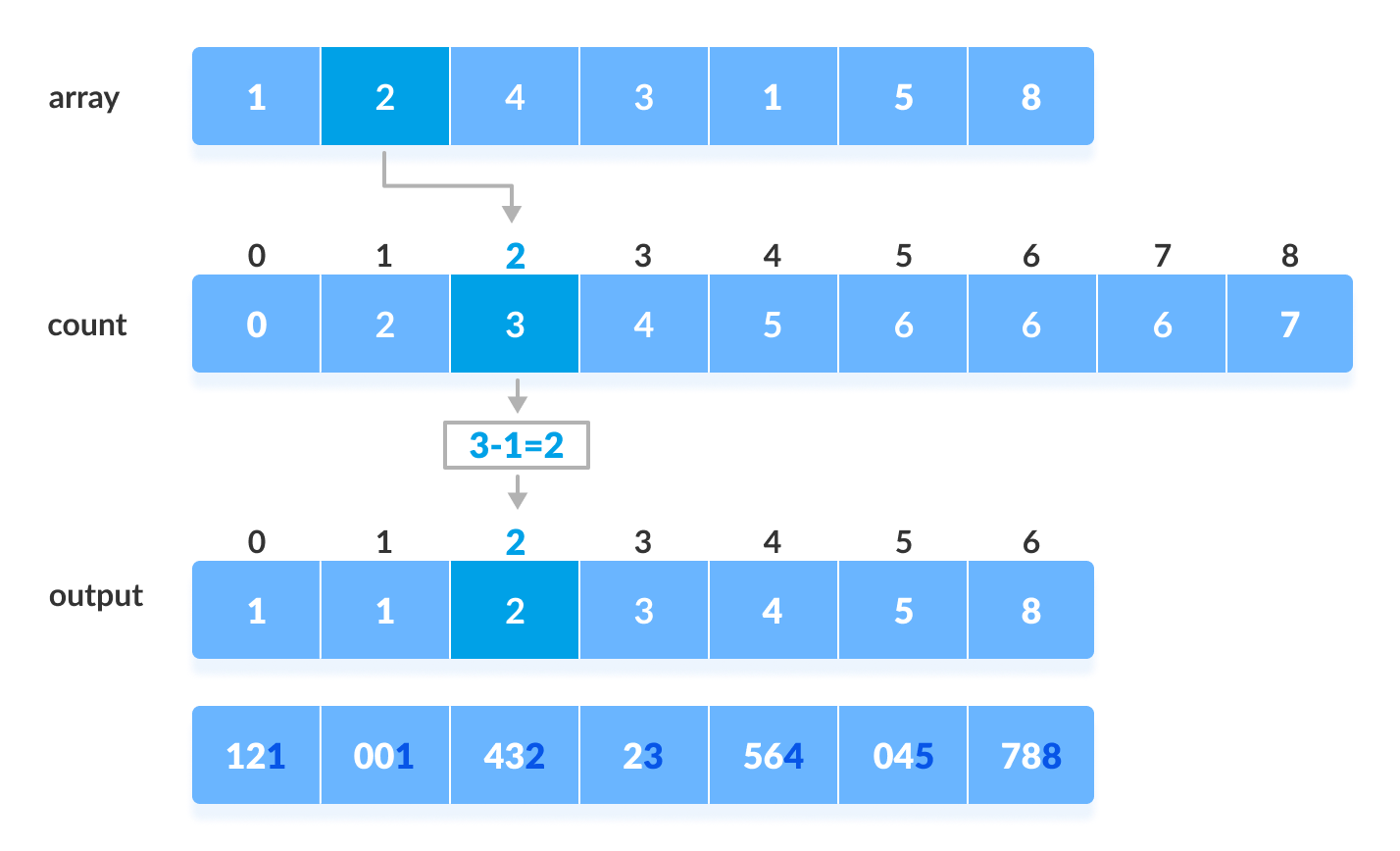
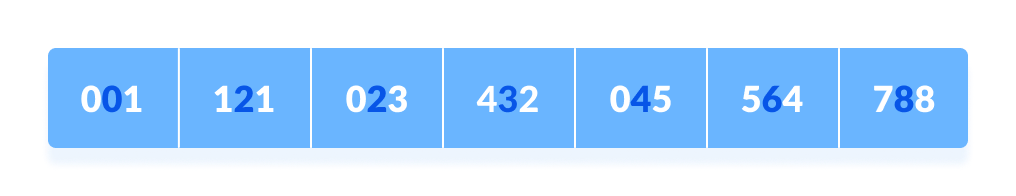
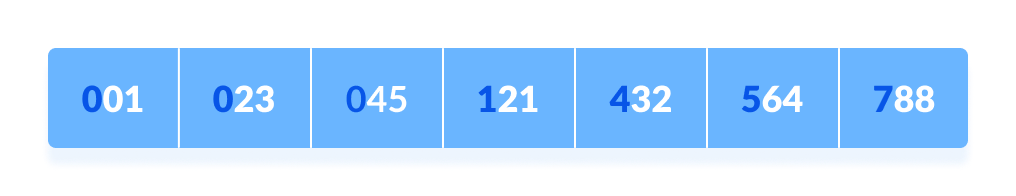
İlk dizi olsun [121, 432, 564, 23, 1, 45, 788]. Aşağıdaki şekilde gösterildiği gibi radix sıralamasına göre sıralanır.



Radix Sort'un Çalışması

Lütfen bu makaleyi okumadan önce [sayma sıralamasını](https://www.programiz.com/dsa/counting-sort) gözden geçirin çünkü sayım sıralaması, sayı tabanı sıralamasında bir ara sıralama olarak kullanılır.

**Radix Sort'un Çalışması**

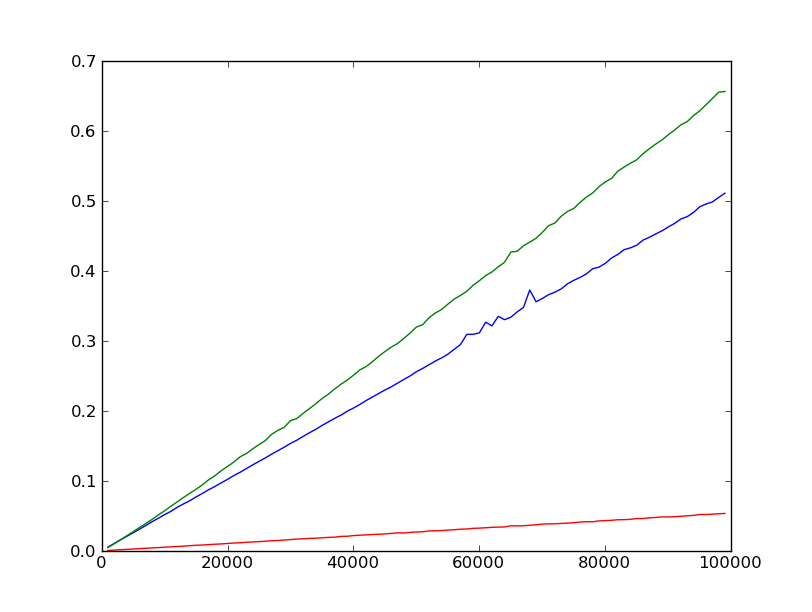
1. Dizideki en büyük elemanı bulun, yanimaksimum. Xiçindeki rakam sayısı olsun max. Xhesaplanır çünkü tüm öğelerin tüm önemli yerlerinden geçmemiz gerekir.  
     
   Bu dizide [121, 432, 564, 23, 1, 45, 788]en büyük sayı 788'e sahibiz. 3 rakamı var. Bu nedenle döngü yüzlerce basamağa (3 kez) çıkmalıdır.
2. Şimdi, her önemli yeri tek tek gözden geçirin.  
     
   Her önemli yerdeki rakamları sıralamak için herhangi bir kararlı sıralama tekniğini kullanın. Bunun için sayma sıralamasını kullandık.  
     
   Öğeleri, birim basamak basamaklarına ( X=0) göre sıralayın.Öğeleri birim yere göre sıralamak için sayma sıralamasını kullanma
3. Şimdi, öğeleri onlar basamağında rakamlara göre sıralayın.Öğeleri onlarca yere göre sıralama
4. Son olarak, öğeleri yüzlerce basamaktaki rakamlara göre sıralayın.Öğeleri yüzlerce yere göre sıralayın

**Radix Sıralama Uygulamaları**

Radix sıralama şurada uygulanır:

* Bir sonek dizisi yaparken DC3 algoritması (Kärkkäinen-Sanders-Burkhardt).
* geniş aralıklarda sayıların olduğu yerler.

**Boyut-Zaman Grafikleri**



(Kırmızı: 10’luk, Mavi: 100’lük, Yeşil: 1000’lik)

Radix sıralama karşılaştırmalı olmayan bir algoritma olduğundan, karşılaştırmalı sıralama algoritmalarına göre avantajları vardır. Bu nedenle yukarıda da görüldüğü gibi Radix Sort daha verimlidir.

**Radix Sort Algoritmalarının Karmaşıklıkları**

|  |  |
| --- | --- |
| **Zaman Karmaşıklığı** |  |
| En iyi | O(n+k) |
| En kötüsü | O(n+k) |
| Ortalama | O(n+k) |
| **Uzay Karmaşıklığı** | O(maks) |
| **istikrar** | Evet |

Radix sıralama karşılaştırmalı olmayan bir algoritma olduğundan, karşılaştırmalı sıralama algoritmalarına göre avantajları vardır.

Bir ara kararlı sıralama olarak sayma sıralamasını kullanan sayı tabanı sıralaması için zaman karmaşıklığı O(d(n+k)).

Burada, dsayı döngüsü ve O(n+k)sayma sıralamasının zaman karmaşıklığıdır.

O(nlog n)Bu nedenle, sayı tabanı sıralama, karşılaştırmalı sıralama algoritmalarından daha iyi olan doğrusal zaman karmaşıklığına sahiptir .

Çok büyük basamaklı sayıları veya 32-bit ve 64-bit sayılar gibi diğer tabanların sayısını alırsak, o zaman doğrusal zamanda çalışabilir, ancak ara sıralama büyük yer kaplar.

Bu, sayı tabanı sıralama alanını verimsiz hale getirir. Bu sıralamanın yazılım kitaplıklarında kullanılmamasının nedeni budur.