

Öncelik Kuyrukları (Priority Queues)

Dr. Hakan TEMİZ

Öncelik Kuyrukları

- Öncelik kuyruğu, koleksiyondaki elemanları bir öncelik kriterine (anahtar) göre sıralı tutar.
- İstendiğinde de, öğeleri önceliklerine göre verir.
- En küçük anahtara sahip öğe en yüksek önceliğe sahipse (ilk önce en küçük öğeyi verir) artan öncelikli kuyruk; aksi durumda ise azalan öncelikli kuyruk (ilk önce en büyük öğeyi verir) denir.
- Öncelik kuyruğu, tipik kuyruğa benzemekle beraber; bazı farklılıklar arz eder.
- Tipik bir kuyrukta elemanlar ilk giren ilk çıkar (FIFO) yapısına sahip iken; öncelik kuyruğunda, elemanların kuyruğu alınış sırasının bir önemi yoktur.
- Öğeler herhangi bir sıra ile gelebilir fakat eleman silinmek istendiğinde, artan öncelikli kuyrukta en büyük anahtar değerli öğe ilk önce; azalan öncelikli kuyrukta ise en küçük anahtar değerli öğe ilk önce kuyruktan çıkarılır.

Öncelik Kuyrukları

- Öncelik kuyrularının kullanımına işletim sistemlerinde görevlerin önceliklendirilmesi gösterilebilir. İşlemciye gelen işler belirli bir öncelik kriterine göre öncelik kuyruğuna alınır ve en yüksek öncelikli işler ilk önce işlemciye gönderilir.
- Öncelik kuyruklarının daha birçok uygulaması vardır. Bunların bazıları şunlardır:
 - Veri sıkıştırma: Huffman Kodlama algoritması
 - Seçim problemi: k'inci en küçük elemanı bulma
 - En kısa yol algoritmaları: Dijkstra algoritması
 - Minimum yayılan ağaç algoritmaları: Prim algoritması

Öncelik Kuyrukları - İşlemler

Tipik bir öncelik kuyruğunda asgari şu işlemler bulunur:

- **Ekle (anahtar, veri):** Öncelik kuyruğuna anahtarlı verileri ekler. Öğeler anahtara göre sıralanır.
- **EnKucukSil (DeleteMin) / EnBuyukSil (DeleteMax):** En küçük/en büyük anahtara sahip öğeyi kuyruktan çıkarır ve verir.
- **EnKucuk (GetMinimum) / EnBuyuk (GetMaximum):** En küçük/en büyük anahtara sahip öğeyi silmeden verir (döndürür).

Öncelik Kuyrukları

Öncelik kuyrukları şu veri yapıları kullanılarak gerçekleştirilebilir:

- **Diziler.** Dizilerde gerçekleştirim, sıralı veya sırasız biçimde gerçekleştirilebilir. Sıralı gerçekleştirimde elemanlar önceliklerine göre dizide sıraya eklenirken; sırasızda gelişlerine göre sıraya eklenirler. Sırasız olması durumunda en öncelikli eleman bulunur ve diziden çıkarılır.
- **Bağlı Listeler.** Bağlı listeler ile gerçekleştirim de, dizilerdeki gibi sıralı veya sırasız olabilir. Sıralı gerçekleştirimde, elemanlar listeye önceliklerine göre bağlanır. Diğer durumda gelişlerine göre eklenir ve en öncelikli eleman istendiğinde arama işlemi neticesinde bulunarak verilir, listeden çıkarılır.
- **İkili Arama Ağaçları.** İkili arama ağaçları ile gerçekleştirimde, ekleme ve çıkarma işlemleri ortalama olarak $O(\log n)$ zaman alır. Dengeli ikili arama ağaçları da gerçekleştirim için kullanılır.
- **İkili Kümeleme Ağaçları.** İkili kümeleme ağaçlarında ekleme, silme ve arama işlemleri $O(\log n)$; minimum ve maksimum değeri bulma $O(1)$ zaman alır.

Öncelik Kuyrukları - Karmaşıklık

● Gerçekleştirim	Ekleme	Silme	Arama (En Küçüğü)
● Sırasız Dizi	1	n	n
● Sırasız Bağlı Liste	1	n	n
● Sıralı Dizi	n	1	1
● Sıralı Bağlı Liste	n	1	1
● İkili Arama Ağacı	$\log n$ (ortalama)	$\log n$ (ortalama)	$\log n$ (ortalama)
● Dengeli İkili Ağaç	$\log n$	$\log n$	$\log n$
● İkili Kümeleme Ağacı	$\log n$	$\log n$	1