

Birleřtirme Sıralaması ve Hızlı Sıralama Nedir?

Çalışma Kağıdı

Birleřtirme Sıralaması, "böl ve yönet" prensibini kullanarak bir diziyi özyinelemeli olarak alt dizilere ayırıp sonra bunları sıralı bir şekilde birleřtirirken; Hızlı Sıralama, bir "pivot" eleman seçerek diziyi bu pivot etrafında iki alt diziyeye ayırır ve bu işlemi özyinelemeli olarak devam ettirir.

Sorular

1. Ařağıdakilerden hangisi Birleřtirme Sıralaması'nın birleřtirme adımında kullanılır?

- A) Pivot seçimi
- B) İki sıralı alt diziyi tek bir sıralı diziyeye birleřtirmek
- C) Rastgele eleman seçimi
- D) Diziyi sabit elemanlara göre bölmek

2. Hızlı Sıralama'nın en kötü durum zaman karmařıklığına neden olan durum nedir?

- A) Pivotun her zaman ortadaki eleman olması
- B) Pivotun her zaman en küçük veya en büyük eleman olması
- C) Dizinin zaten sıralı olması
- D) Dizinin ters sıralı olması

3. Hangi sıralama algoritması genellikle daha fazla ek bellek alanı gerektirir?

- A) Birleřtirme Sıralaması
- B) Hızlı Sıralama
- C) Kabarcık Sıralaması
- D) Seçmeli Sıralama

4. Birleřtirme Sıralaması ile [38, 27, 43, 3, 9, 82, 10] dizisini sıralayın.

5. Hızlı Sıralama ile [38, 27, 43, 3, 9, 82, 10] dizisini sıralayın (pivot olarak son elemanı seçelim).

6. Tanımla: Birleřtirme Sıralaması'nın temel prensibi nedir?

7. Tanımla: Hızlı Sıralama'da 'pivot' ne işe yarar?

8. Tanımla: Birleřtirme Sıralaması'nın zaman karmařıklığı nedir (en kötü durum)?

Cevap Anahtarı

1. B) İki sıralı alt diziyi tek bir sıralı diziyeye birleştirmek — Birleştirme Sıralaması'nın ana adımı, iki adet sıralanmış alt diziyi alıp bunları tek bir büyük sıralanmış dizi haline getirmektir.
2. B) Pivotun her zaman en küçük veya en büyük eleman olması — Hızlı Sıralama'da pivot seçimi kritiktir. Eğer pivot sürekli olarak dizinin en küçük veya en büyük elemanı olarak seçilirse, bölme işlemi dengesiz olur ve karmaşıklık $O(n^2)$ 'ye çıkar.
3. A) Birleştirme Sıralaması — Birleştirme Sıralaması, alt dizileri birleştirirken geçici olarak ek bellek alanına ihtiyaç duyar (genellikle $O(n)$ alan karmaşıklığına sahiptir). Hızlı Sıralama ise genellikle yerinde (in-place) sıralama yapar ve daha az ek bellek kullanır.
4. Dizi ikiye bölünür: [38, 27, 43, 3] ve [9, 82, 10], Her alt dizi tekrar bölünür ve sıralanır: [27, 38, 43] ve [3] ile [9, 10, 82], Sıralanmış alt diziler birleştirilir: [3, 27, 38, 43] ve [3, 9, 10, 82], Son olarak, bu iki sıralı dizi birleştirilir: [3, 9, 10, 27, 38, 43, 82]
5. Pivot: 10. Diziyi 10 etrafında ayır: [3, 9] (küçükler) ve [38, 27, 43, 82] (büyükler). Dizi: [3, 9, 10, 38, 27, 43, 82], Sol alt dizi [3, 9] için pivot: 9. Ayır: [3] (küçük) ve [] (büyük). Dizi: [3, 9], Sağ alt dizi [38, 27, 43, 82] için pivot: 82. Ayır: [38, 27, 43] (küçükler) ve [] (büyükler). Dizi: [38, 27, 43, 82], Tekrar eden adımlarla tüm alt diziler sıralanır ve birleştirilir: [3, 9, 10, 27, 38, 43, 82]
6. Böl ve Yönet (Divide and Conquer)
7. Diziyi pivot etrafında iki alt diziyeye ayırarak sıralama işlemini basitleştirir.
8. $O(n \log n)$

Bounlu

Tüm kartlar, adım adım çözümler ve AI hoca desteği Notek uygulamasında.
Sınav tarihlerini Promy otomatik hatırlatıcıya çevirir.