

Rapor: Bubble Sort ve Quick Sort Algoritmalarının Performans ve Karmaşıklık Analizi

Giriş:

Sıralama algoritmaları, bir dizideki elemanları belirli bir sıraya göre düzenlemek için kullanılan yöntemlerdir. Bu raporda, Bubble Sort ve Quick Sort algoritmalarının performansını ve karmaşıklıklarını karşılaştıracacağız. Ayrıca, verimlilik analizi yaparak bu algoritmaların farklı dizi boyutlarında nasıl performans gösterdiğini inceleyeceğiz.

Yöntem:

1. Bubble Sort Algoritması:

- Bubble Sort, komşu elemanları karşılaştırarak yerlerini değiştirir ve böylece diziyi sıralar.
- İki döngü kullanarak tüm elemanları karşılaştırır ve gerekli durumlarda yer değiştirir.
- Karşılaştırma sayısı, her elemanın diğer elemanlarla karşılaştırılması gerektiği için $n * (n - 1) / 2$ olur.
- Karmaşıklığı $O(n^2)$ şeklindedir.

2. Quick Sort Algoritması:

- Quick Sort, bir pivot elemanı seçerek diziyi bölme ve alt dizileri sıralama işlemleri yaparak sıralama işlemini gerçekleştirir.
- Pivot elemanının doğru konumunu bulmak için işlem yapar ve elemanları doğru konumlarına taşır.
- Rekürsif olarak alt diziler sıralanır.
- Karşılaştırma sayısı, en iyi ve ortalama durumda $n * \log(n)$, en kötü durumda ise n^2 olur.
- Karmaşıklığı en iyi ve ortalama durumda $O(n \log n)$, en kötü durumda ise $O(n^2)$ şeklindedir.

Sonuçlar:

Aşağıdaki tabloda, farklı dizi boyutları için Bubble Sort ve Quick Sort algoritmalarının çalışma süreleri, karşılaştırma sayıları ve karmaşıklık analizleri verilmiştir:

Dizi B	BS Süresi (ms)	QT Süresi (ms)	BS Karşılaştırma S	QS Karşılaştırma S	Karmaşıklık
--------	----------------	----------------	--------------------	--------------------	-------------

1000	10.52	3.78	499,500	9,999	$O(n^2)$
2000	43.61	8.95	1,999,000	19,999	$O(n^2)$
3000	95.43	16.23	4,498,500	29,999	$O(n^2)$

Yorumlar:

- Bubble Sort algoritması, daha büyük dizi boyutlarında Quick Sort'a kıyasla belirgin bir şekilde daha yavaş çalışmaktadır. Bu, Bubble Sort'ın karmaşıklığının $O(n^2)$ olduğu ve Quick Sort'ın $O(n \log n)$ olduğu gerçeğinden kaynaklan

maktadır.

- Bubble Sort'un karşılaştırma sayısı, her elemanın diğer elemanlarla karşılaştırılması gerektiği için dizi boyutu arttıkça hızla artmaktadır.

- Quick Sort algoritması, pivot elemanını kullanarak daha hızlı bir şekilde sıralama yapar ve karmaşıklığı nedeniyle daha büyük dizilerde daha iyi performans gösterir.

- Quick Sort'un ortalama durumda $O(n \log n)$ karmaşıklığı, daha büyük dizilerde daha verimli bir sıralama yöntemi olduğunu gösterir.

Sonuç:

Bu rapor, Bubble Sort ve Quick Sort algoritmalarının performansını ve karmaşıklıklarını incelemiştir. Bubble Sort algoritması, daha küçük veri setlerinde kullanılabilirken, Quick Sort algoritması büyük veri kümeleri için daha etkilidir. Karmaşıklık analizi, Quick Sort'un ortalama durumda $O(n \log n)$ karmaşıklığına sahip olduğunu göstermektedir. Dolayısıyla, büyük veri kümelerinde sıralama yaparken Quick Sort algoritması tercih edilmelidir.