



Algoritma Analizi Ödev -2

Öğrenci Adı: Ömer Diner

Öğrenci Numarası: 20011017

Dersin Eğitmeni: M. Amaç Güvensan

Video Linki:

https://drive.google.com/file/d/1Z4SUGWlt6QzhZ_kPZoaYVMkXXACTp_Zs/view?usp=sharing

1- Problem Tanımı:

Merge sort algoritmasının bir farklı versiyonu olan k-yollu birleştirme algoritması ile veri dizilerinin sıralanması işlemi yapılacaktır. Böl ve yönet(divide & conquer) yaklaşımı kullanılarak tasarlanan bu algoritma, belirli bir k değeri ile birleştirme işlemini gerçekleştirir. Algoritmanın performansı, farklı boyutlardaki veri setleri üzerinde ve değişen k değerleri için analiz edilecektir. Deneysel sonuçların, farklı N ve k değerlerine göre çalışma sürelerini grafik yardımıyla karşılaştırmalı olarak sunulması istenmektedir. N = 100, 1.000, 10.000, 100.000, 1.000.000 ve 10.000.000 elemanlı diziler üzerinde k=2 ile k=10 arasındaki değerler kullanılarak testler yapılacaktır.

2- Problemin Çözümü:

Kod aşağıdaki adımlarla tasarlanıp uygulanmıştır:

Dizilerin Rastgele Oluşturulması: Belirli N değerlerine göre rastgele ve tekil elemanlar içeren diziler üretilmiştir. Elemanlar, 1 ile N arasında olacak şekilde üretilmiş ve tüm elemanların benzersiz olması sağlanmıştır. Bu dizi her k değeri için aynı kalacak şekilde saklanmıştır, böylece farklı k değerleri için aynı verilerle test yapılmıştır.

K-way merge sort: Dizi, k parçaya bölündükten sonra, her bir parça kendi içinde sıralanır. Sıralanan k parçanın elemanları uygun şekilde birleştirilerek sıralı bir dizi oluşturulur. Bu işlem, böl ve yönet yaklaşımını kullanarak k yola bölünmeyi içerir. Böylece k farklı parçanın aynı anda birleştirilmesi sağlanır.

Çalışma Süresi Ölçümü: clock() fonksiyonu ile her N ve k değeri için algoritmanın çalışma süresi ölçülmüştür.

Sonuçların Dosyaya Kaydedilmesi: Ölçülen süreler ve kullanılan parametreler, analiz edilmek üzere dosyaya kaydedilmiştir.

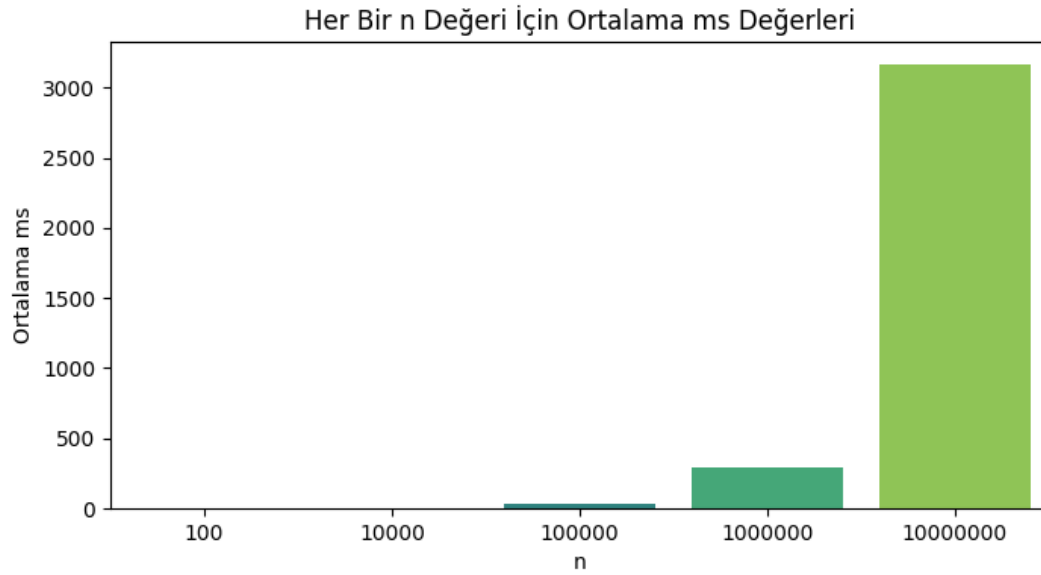
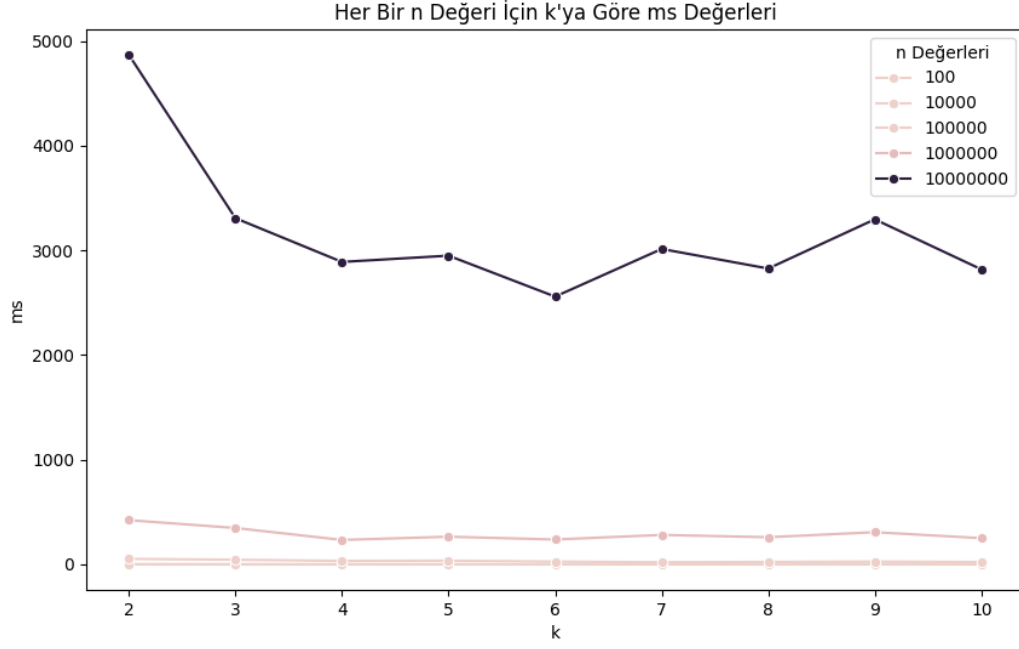
Deneysel Analiz: Elde edilen süreler farklı N ve k değerleri için Python kullanılarak grafikler halinde görselleştirme yapılmıştır. Bu karşılaştırmalar sayesinde, dizinin boyutunun ve k değerinin algoritmanın performansı üzerindeki etkisi incelenmiştir.

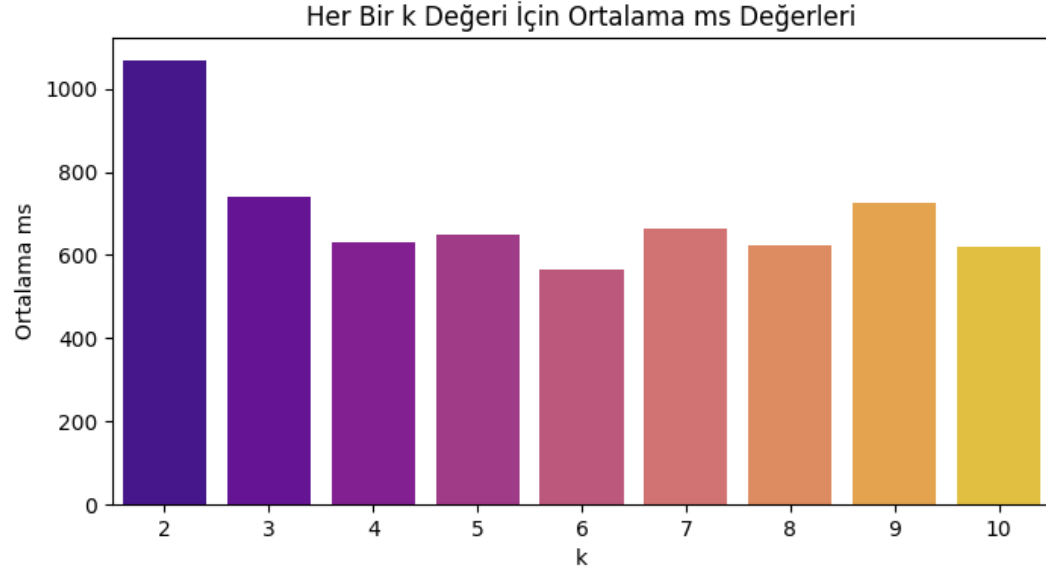
3- Karşılaşılan Sorunlar:

Eleman sayısı k'dan küçük olan durumlar için insertion sort kullanıldı.

Her elemanı farklı dizi oluştururken eklenen eleman daha önce eklendi mi diye kontrol koyulmaması amacıyla ilk önce n adet 1 .. n sayı üretildi. Sonra bu sayılar kendi arasında yerleri değiştirilerek istenen dizi elde edildi.

4- Deneysel Analiz ve Çıktılar:





Dizi eleman sayısının algoritmanın çalışma süresini artırdığı gözlemlenmiştir.

K yüksek değerlerde daha fazla alt dizi oluşturduğu için çalışma süresini arttırabiliyor ancak düşük k değerleri de birleştirme işleminde daha fazla veri ile işlemin yapılmasına yol açtığı için süreyi uzatıyor.

En düşük performans k değeri 2 iken alınmıştır.

Bu sebeple k değeri 6 değeri orta bir değer olarak ortalamada en hızlı sonuçları vermiştir.

100 ve 10.000 elemanlı veriler üzerinde yapılan çalışmalar çok hızlı tamamlandığı için sonuçlarda 0 ms olarak görüldü.

Veri setinin boyutuna göre bu tarz deneyler tekrarlanarak dikkatli bir şekilde k değeri seçilerek en hızlı çalışma süresi elde edilebilir.