

# Sıralama Algoritmaları Görselleştiricisi

Caner Ay Celep  
201307036  
[201307036@kocaeli.edu.tr](mailto:201307036@kocaeli.edu.tr)

Eyüp Ayaz  
201307024  
[201307024@kocaeli.edu.tr](mailto:201307024@kocaeli.edu.tr)

## Abstract

This project case we have to a sorted algorithm. We added charts for this project. The charts were displayed for shorted algorithm. Users can be display this charts. If users want speed of algorithm they can be change speed.

**Keywords**— algorithm, charts, shorted, speed, insert (key words)

## I. KONUSU

Proje kapsamında sıralama algoritmalarına görsel arayüz ile kullanıcıların algoritmaların farkını , zaman karmaşıklıklarını ve görsel olarak şekilleştirici grafikler ile görmeleri için bir uygulama geliştirdik. Bu bağlamda kullanıcılar istedikleri zaman algoritmaları kıyaslayabilir , dilerlerse özel olarak kendi tanımladıkları dizileri girebilir ve bunların sıralanmasını isteyebilirler. Kullanıcılar isterlerse algoritmaların yavaşlık ve hızlık ayarlarını scrollbar sayesinde ayarlayabilmektedir.

### A. Anasayfa

Projemizi .netcore 5.0 kullanarak gerçekleştirdik. Projemizde herhangi bir platform bağımlılığı olmaması için böyle bir yol izlemiş olduk. Projemiz bir masaüstü uygulaması olarak karşımıza çıkmakta. Kullanıcılar istedikleri algoritmanın ilgili butonlarına bastıkları zaman zaman karmaşıklıkları , toplam ne kadar sürede gerçekleştiği gibi değişkenleri görebilirler. İlgili uygulamamızın ekran görüntüsü “Şekil-1”’de gösterilmiştir.



Şekil-1

### B. İşleyiş

Sıralama algoritmalarımız için methodları tanımlamış olduk. Selection, Buble, Insertion, Merge, Quic Sort algoritmalarımız için tanımlanmış olan methodlara kullanıcı ilgili radio butona basması durumunda anlık olarak zaman karmaşıklığı, uzun karmaşıklığı ve geçen süre gibi bilgiler

kullanıcıya gösterilmektedir. Kullanıcı bu sayede ilgili algoritmaların radio butonlarına basarak ilgili işlemleri gerçekleştirebilir ve kıyaslama işlemlerini yapabilir. İlgili ekran çıktımız “Şekil-2”’de gösterilmektedir.



Şekil-2

## II. KULLANILAN GRAFİKLER

Projemizde kullanmış olduğumuz algoritmaları kullanıcıların görsel olarak daha iyi görebilmesi ve anlaşılabilir olması için çeşitli grafik türlerinden yararlandık. Bar, Scatter, Stem grafikleri sayesinde kullanıcılar istediği grafik türlerini seçip algoritmaların görsel olarak yansımalarını görebilirler ve bu şekilde ayrımları yapabilirler. Kullanıcılar grafiğin hızlık ayarını istedikleri zaman uygulamada messagebox kullanarak hazırlanmış olduğumuz diyalog penceresi ile durdurup hızını değiştirebilir bu sayede daha hızlı veya yavaş bir görünüm elde edilebilir. İlgili devam etme ve durma menüsü “Şekil-3”’de gösterilmiştir.



Şekil-3

## A. BAR

Kullanıcıların algoritmaları görsel olarak görebilmesi için grafiklerden yararlandığımızdan bahsetmiştik. Bar grafiği dikey çizgiler sayesinde kullanıcılara sıralama algoritmalarında sıralanan değerlerin büyüklük küçüklük değerine göre dikey çubukların boyutlarına bağlı kalarak görsel bir arayüz sunar. Bu sayede daha büyük olan değerlerin dikey çubukları daha yüksekte olur. Kullanıcı bu sayede ayrımı yapabilir. İlgili şekil “Şekil-4” ‘de verilmektedir.



Şekil-4

## B. SCATTER

Scatter grafiğinde ise kullanıcılar verilen değerlerin görsel olarak şeklini serpilme diyagramı şeklinde görmektedir. Bu sayede kullanıcılar ilgili algoritmanın değişkenlerini ve zamana bağlı değişimini grafikte görebilir. Bu sayede kullanıcılar seçmiş oldukları algoritmaların renkli küçük daire şeklinde sıralanmış değerlerini kordinat düzleminde görebilirler aynı zamanda dilerlerse hız ayarlarını değiştirip aradaki varki görsel olarak deneyimleyebilirler. İlgili şekil “Şekil-5” ‘de verilmektedir.



Şekil-5

## C. STEM

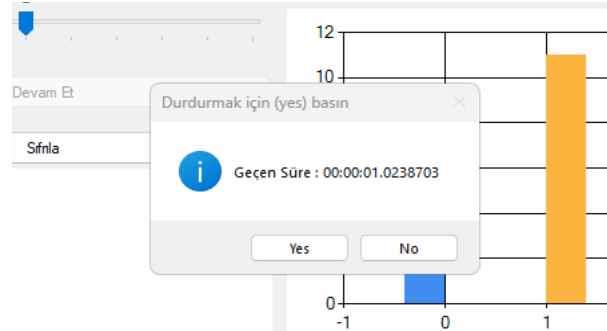
Stem grafiğinde ise kullanıcılar ilgili sıralama algoritmalarını seçtikten sonra stem grafiği sayesinde sıralanan değerleri dal yaprak grafiği olarak

görüntüleyebilir. Bu arada çizge şeklinde olan dal yaprak grafikleri ile sıralanmış oldukları değerlerin ayrımını yapabilir ve görsel olarak bunları deneyimleyebilirler. İlgili şekil “Şekil-6” ‘da verilmiştir.

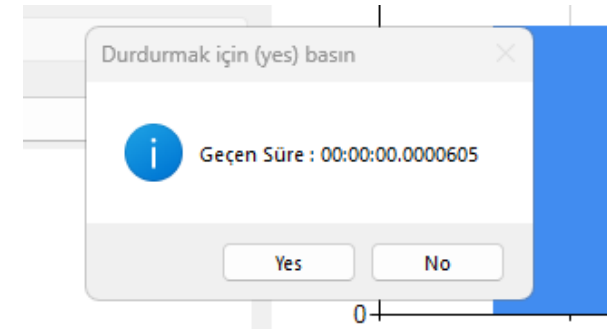


Şekil-6

Kullanıcılar dilerse algoritmanın ve grafiğin işleyişini daha yavaş görmek yani zamanını değiştirmek için ilgili hız ayarımızın mevcut olduğu slider seçeneğini kullanabilir bu sayede seçmiş oldukları algoritmanın hızını değiştirebilir ve algoritmanın arasında geçen süre zarfını görüntüleyebilir. Kullanıcılar seçmiş olduğu algoritmalarındaki değişimleri bu sayede çeşitli grafikler ve bilgilendirmeler aracılığıyla aradaki farkları görmüş oldu. Bu projemizde algoritmaların sürelerine ve hızlara olan etkilerini görsel olarak kullanıcılara bir bildirim uyandırmak için gerçekleştirmiş olduk. Projemizin hız ayarında yapılan değişiklik öncesi ve sonrası “Şekil-7” ve “Şekil-8” ‘de gösterilmiştir.



Şekil-7



Şekil-8

## KAYNAKÇA

- <https://code-maze.com/sorting-algorithms-csharp/>
- <https://sw Harden.com/csdv/plotting-free/microsoft-charting/>
- <https://stackoverflow.com/questions/44829655/how-can-i-plot-more-than-50-000-values-in-a-scatter-chart-saving-computer-resou>