Yazılım Geliştirme Laboratuvarı - II

Sıralama Algoritmaları Görselleştiricisi

Ömer Faruk Yılmaz  *Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi*  
191307048

Sana Kabbani *Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi*  
181307074

Kemal Turak *Bilişim Sistemleri Mühendisliği  
Kocaeli Üniversitesi*  
191307074

*Özetç:* *backend olarak Python frontend olarak PyQt5 ve Matplotlib kütüphaneleri kullanılarak Sırlama Algoritmaları Görselleştirici bir masaüstü uygulaması bir proje hazırlanmıştır*

Anahtar Kelimeler: Python , MasaÜstü , Algoritma , Matplotlib ,PYQT6, Numby, Görselleştırme

# Giriş

Bu rapor, PyQt5, Matplotlib ve numpy gibi Python kütüphanelerini kullanarak bir sıralama algoritması görselleştirme uygulamasının kodunu içermektedir.

Amacı, farklı sıralama algoritmalarının çalışma prensiplerini anlamak ve kullanıcılara bu algoritmaların nasıl işlediğini göstermektir.

Kullandığımız sıralama algoritmaları şunlardır:

* Seçme Sıralaması (Selection Sort) :

Verinin hafızada sıralı tutulması için geliştirilen sıralama algoritmalarından (sorting algorithms) bir tanesidir. Basitçe her adımda dizideki en küçük sayının nerede olduğu bulunur. Bu sayı ile dizinin başındaki sayı yer değiştirilerek en küçük sayılar seçilerek başa atılmış olur.

* Kabarcık Sıralaması (Bubble Sort) :

Verinin hafızada sıralı tutulması için geliştirilen sıralama algoritmalarından (sorting algorithms) bir tanesidir. Basitçe ardışık duran iki hafıza bloğunun birbirine nispetle sıralanması ve bu işlemin sürekli tekrarlanması sayesinde sıralar. Ardışık iki hafıza bloğuna bakmasından dolayı baloncuk ismini almıştır. Çünkü bu bakma işlemi bir baloncuğun (buble) hareket etmesi gibi sayıların üzerinde hareket etmektedir.

* Ekleme Sıralaması (Insertion Sort) :

Sokma sıralaması, programlaması oldukça basit ancak performansı bölme sıralaması (merge sort), hızlı sıralama(quick sort) gibi sıralamalara göre nispeten yavaş bir sıralama algoritmasıdır.

* Birleştirme Sıralaması (Merge Sort) :

Verinin hafızada sıralı tutulması için geliştirilen sıralama algoritmalarından (sorting algorithms) bir tanesidir. Basitçe sıralanacak olan diziyi ikişer elemanı kalan parçalara inene kadar sürekli olarak ikiye böler. Sonra bu parçaları kendi içlerinde sıralayarak birleştirir. Sonuçta elde edilen dizi sıralı dizinin kendisidir. Bu açıdan bir parçala fethet (divide and conquere) yaklaşımıdır.

* Hızlı Sıralama (Quick Sort) :

Basitçe sıralanacak olan dizideki orta noktada (mean) bulunan bir sayıyı seçerek diğer bütün sayıları bu orta sayıdan büyük veya küçük diye sınıflayarak sıralama yapmayı hedeflemektedir. Bu açıdan bir parçala fethet (divide and conquere) yaklaşımıdır. Ayrıca bu seçilen orta noktaya eksen (pivot) adı da verilir çünkü bütün diğer sayılar bu sayının ekseninde sıralanacaktır.

Kod, PyQt5 arayüz kütüphanesini kullanarak bir kullanıcı arayüzü oluşturur. Bu arayüzde, kullanıcı veri kümesini oluşturabilir, sıralama algoritmasını seçebilir ve görselleştirme türünü belirleyebilir. Kullanıcı ayrıca sıralama işlemini başlatabilir ve durdurabilir. Algoritma çalıştığında, her adımda grafik güncellenir ve kullanıcıya gösterilir.

Bu raporda, proje geliştirme süreci, kullanılan teknolojiler ve proje sonuçları ayrıntılı olarak ele alınacaktır.

# Kullanılan teknolojiler

Uygulama, Python programlama dili ve çeşitli Python kütüphanelerini kullanmaktadır.

## Back-End Teknolojileri

1. Python: Uygulama, Python programlama diliyle geliştirilmiştir. Python, basit ve anlaşılır bir sözdizimine sahip, geniş kütüphane desteği olan bir programlama dilidir.
2. numpy: numpy, Python'da bilimsel hesaplama ve veri işleme için kullanılan bir kütüphanedir. Uygulama, numpy'ı veri kümesini oluşturmak ve manipüle etmek için kullanır.

## Front-End Teknolojileri

a. PyQt5: PyQt5, Python için bir GUI (Grafik Kullanıcı Arayüzü) kütüphanesidir. Uygulama, PyQt5'i kullanarak kullanıcı arayüzünü oluşturur ve etkileşim sağlar

. b. Matplotlib: Matplotlib, Python'da veri görselleştirme için kullanılan popüler bir kütüphanedir. Uygulama, Matplotlib'i kullanarak veri kümesini grafiksel olarak görselleştirir.sağlamak ve verimli UI (kullanıcı arayüzü) oluşturmak için kullanılır

# Veri Kümesi

Veri kümesi, kullanıcı tarafından girdi olarak sağlanan veya rastgele oluşturulan sayısal değerlerin bir listesini temsil eder. Bu sayılar, sıralama algoritmalarıyla işlenir ve grafik üzerinde görselleştirilir.

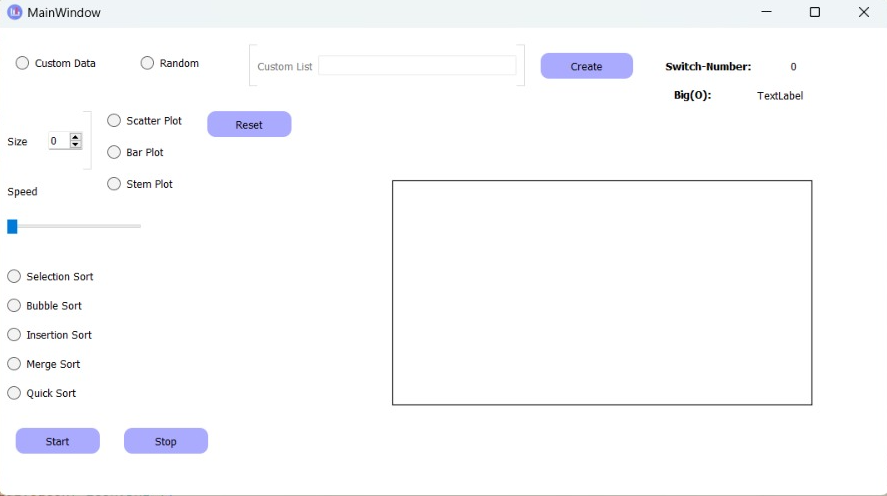
Kod içerisinde, veri kümesi iki farklı şekilde oluşturulabilir:

1. Rastgele Veri Kümesi: "Random" olarak işaretlenmiş radyo düğmesi seçildiğinde, kullanıcı tarafından belirtilen aralıkta rastgele sayılar üretilir. Bu sayılar, çubuk grafiği üzerinde görselleştirilir.
2. Özel Veri Kümesi: "Custom" olarak işaretlenmiş radyo düğmesi seçildiğinde, kullanıcı tarafından metin kutusuna girilen sayılar kullanılır. Girilen sayılar, boşluk karakteriyle ayrılmış bir dizeden ayrıştırılarak bir liste oluşturulur. Bu sayılar, çubuk grafiği üzerinde görselleştirilir.

# front-end içeriği ve açıklaması

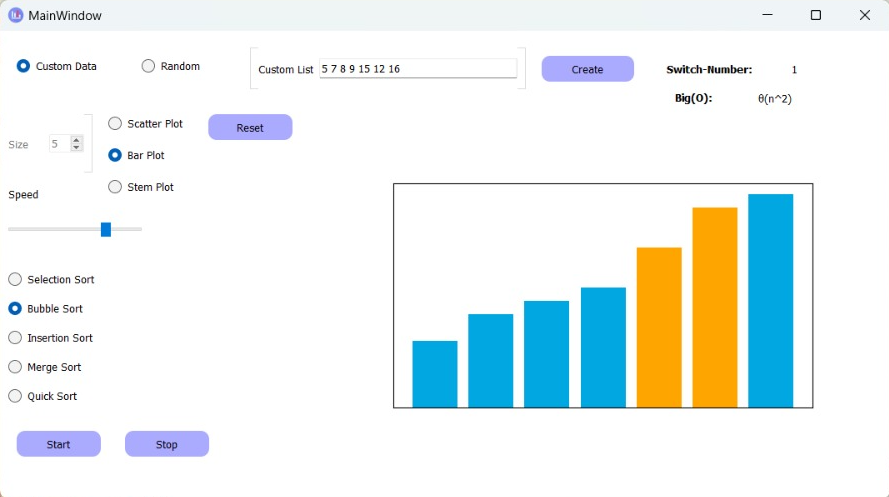
PyQt6, Python dilinde kullanılan bir GUI (Grafiksel Kullanıcı Arayüzü) kütüphanesidir. Qt framework'üne dayanır ve grafiksel uygulamalar geliştirmek için kullanılır. PyQt6, birçok UI bileşeni ve özelleştirme seçenekleri sunar. Qt Designer adlı araç, kullanıcı arayüzünü sürükle-bırak yöntemiyle tasarlamanıza yardımcı olur.

# Projemize Ait Ekran Görüntüleri



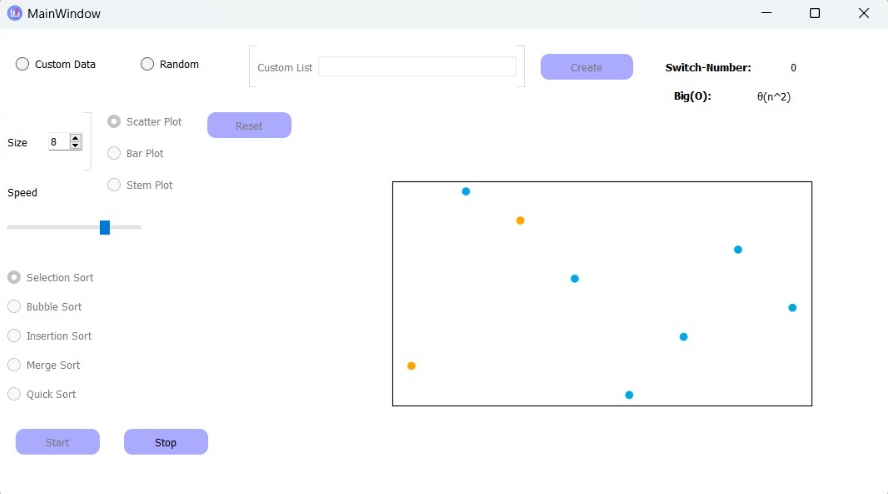
*Şekil 1.0. : Uygulama Arayüzümüzün Görünümü*

PyQt5 grafik kullanıcı arayüzü kütüphanesi ile oluşturmuş olduğumuz ekranımız şekil 1.0.’da gösterilmiştir.



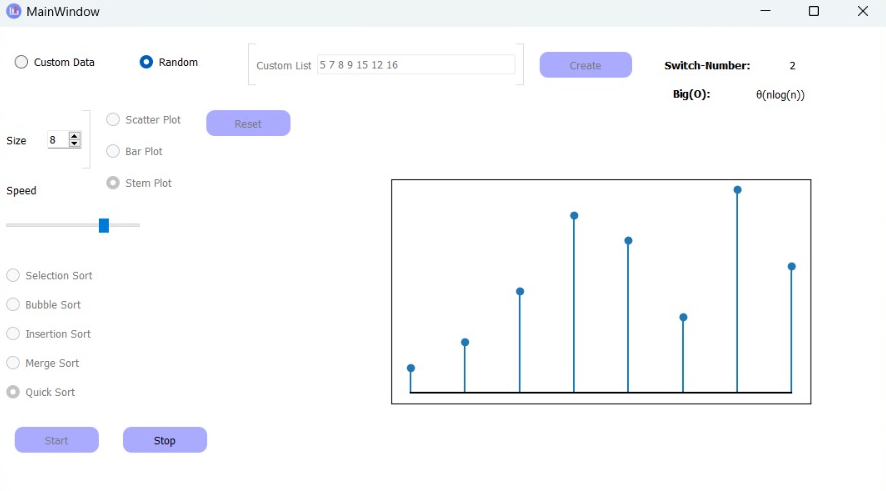
*Şekil 2.0. : Sütun Grafiği*

Manuel olarak yani özel (custom) olarak gönderdiğimiz verilerimizin sütun grafiği olarak uygulamamız üzerindeki görüntüsü şekil 2.0.’da gösterilmiştir.



*Şekil 3.0. : Dağılım Grafiği*

Verilerimizin uygulamamızda dağılım grafiği olarak göründüğü ekran görüntümüz şekil 3.0.’da gösterilmiştir.



*Şekil 4.0. : Kök Grafiği*

Verilerimizin uygulamamızda kök grafiği olarak göründüğü ekran görüntümüz şekil 4.0.’da gösterilmiştir.

# Referanslar

* [https://www.youtube.com/watch?v=IRkvlqPBqNg&t=471s](https://www.youtube.com/watch?v=IRkvlqPBqNg&t=471s%20)
* [https://www.youtube.com/watch?v=RxGlB9U64fg](https://www.youtube.com/watch?v=RxGlB9U64fg%20)
* [https://www.youtube.com/watch?v=RfXt\_qHDEPw&t=179s](https://www.youtube.com/watch?v=RfXt_qHDEPw&t=179s%20)
* [https://github.com/Spoonzio/Sort-and-Shuffle](%20https://github.com/Spoonzio/Sort-and-Shuffle%20)
* [https://www.pythonguis.com/tutorials/plotting-matplotlib/](https://www.pythonguis.com/tutorials/plotting-matplotlib/%20)
* <https://www.pythonguis.com/pyqt6-tutorial/>
* <https://bilgisayarkavramlari.com/2008/08/09/siralama-algoritmalari-sorting-algorithms/>