



SIRALAMA ALGORİTMALARI GÖRSELLEŞTİRME PROJESİ

Aslı Bozkurt, Mohammad Nazir Sharifi, Judy Ndotoni Nkwama

Bilişim Sistemleri Mühendisliği

Kocaeli Üniversitesi

Umuttepe Kampüsü, 41380, Kocaeli-Türkiye

{201307025, 191307094, 191307093}@kocaeli.edu.tr

Abstract- Sorting algorithms is an important topic in the field of programming and data structure. This project provides a user interface that makes it easy to visually understand how various sorting algorithms work. This interface is designed to help students and software developers better understand sorting algorithms. The purpose of this project is to develop an application that visually describes sorting algorithms. The following objectives have been set for this purpose: To enable users to choose different sorting algorithms, Visualizing the step-by-step working process of the selected algorithm, Presenting the number of comparisons and complexity analysis of algorithms, Allowing the user to customize the dataset size and speed, Giving the user the option to enter data manually.

Keywords— datastructures, short algorithms, tkinter, gui

Özet—Sıralama algoritmaları, programlama ve veri yapısı alanında önemli bir konudur. Bu proje, çeşitli sıralama algoritmalarının nasıl çalıştığını görsel olarak anlamayı kolaylaştıran bir kullanıcı arayüzü sunmaktadır. Bu arayüz, öğrencilerin ve yazılım geliştiricilerin sıralama algoritmalarını daha iyi anlamalarına yardımcı olmak için tasarlanmıştır. Bu projenin amacı, sıralama algoritmalarını görsel olarak anlatan bir uygulama geliştirmektir. Aşağıdaki hedefler bu amaç doğrultusunda belirlenmiştir: Kullanıcıların farklı sıralama algoritmalarını seçebilmesini sağlamak, Seçilen algoritmanın adım adım çalışma sürecini görselleştirmek, Algoritmaların karşılaştırma sayısını ve karmaşıklık analizini sunmak, Kullanıcının veri kümesi boyutunu ve hızını özelleştirebilmesini sağlamak, Kullanıcıya el ile veri girişi yapma seçeneği sunmak .

AnahtarKelimeler— veri yapıları, sıralama algoritmaları, uygulama,tkinter, gui

I. GİRİŞ

Sıralama algoritmaları, programlama ve veri yapısı alanında önemli bir konudur. Bu proje, çeşitli sıralama algoritmalarının nasıl çalıştığını görsel olarak anlamayı kolaylaştıran bir kullanıcı arayüzü sunmaktadır. Bu arayüz,

öğrencilerin ve yazılım geliştiricilerin sıralama algoritmalarını daha iyi anlamalarına yardımcı olmak için tasarlanmıştır..

II. PROBLEM

Sıralama algoritmaları, programlama ve veri yapısı alanında önemli bir konudur ve birçok yazılım uygulamasında kullanılmaktadır. Ancak, bu algoritmaların çalışma mantığını ve performanslarını anlamak bazen zor olabilir. Özellikle görsel olarak anlatılmayan durumlarda, öğrencilerin ve yazılım geliştiricilerin bu algoritmaları anlamaları ve kullanmaları zorlaşabilir.

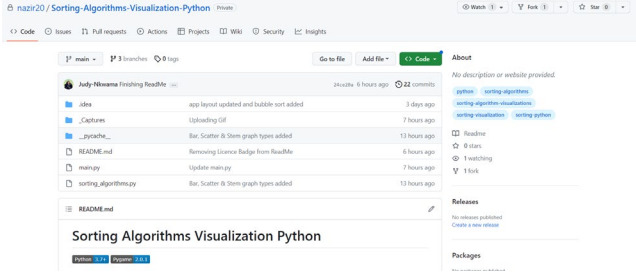
Bununla birlikte, sıralama algoritmalarını görselleştirme imkanı sunan mevcut araçlar ve kaynaklar sınırlıdır. Öğrenciler ve yazılım geliştiriciler, adım adım sıralama algoritmalarının nasıl çalıştığını görsel olarak izlemek ve bu algoritmaların performanslarını karşılaştırmak için daha kullanıcı dostu bir araç arayışındadır.

Bu projenin amacı, sıralama algoritmalarını görsel bir şekilde anlatan bir uygulama geliştirmektir. Bu uygulama, öğrencilere ve yazılım geliştiricilere sıralama algoritmalarını daha iyi anlama ve kullanma fırsatı sunmayı hedeflemektedir. Kullanıcılar, farklı sıralama algoritmalarını seçebilir, bu algoritmaların adım adım çalışma süreçlerini görselleştirebilir ve performanslarını karşılaştırabilirler.

Proje, sıralama algoritmalarını öğrenmek veya mevcut bilgilerini pekiştirmek isteyen öğrenciler, yazılım geliştiriciler ve ilgilenen diğer kişiler için değerli bir araç olacaktır. Ayrıca, projenin kaynak kodu ve görsel dokümantasyonu, sıralama algoritmalarını öğrenmek veya öğretmek için kaynak olarak kullanılabilir.

III. PROJE PLANLAMA

Projede planlama olarak mail ve çevrimiçi uygulamalar ile haberleşme sağlandı. Google Meet ve Discord uygulamaları kullanıldı. Proje geliştirme süreci ve tamamlanması Github üzerinden devam etmiştir.



Resim- 1

IV. KULLANILAN TEKNOLOJİLER

Yazılımı geliştireceğimiz teknolojilere ise gruptaki çoğu kişinin hâkim olduğu ve proje içinde en iyi şekilde geliştirme sağlayacak dil seçimi yaptık. Projenin geliştirilmesinde kullanılan teknolojiler aşağıdaki gibidir.

- PYTHON(BACK-END)
- TKINTER (GUI)

1) *PYTHON (BACK-END)*: Express.js modülü Node.js tabanlı bir web uygulama sunucu çatısıdır. Express.js'in sunduğu sınırsız http yardımcı araçları ve katmanlar sayesinde sağlam bir API oluşturmak oldukça hızlı ve kolaydır. Express.js dinamik web siteleri geliştirmek için kullanılan açık kaynak kodlu yazılım demetidir.

2) *TKINTER (GUI)*: Bu teknolojiyi kullanmamızın amacı uygulamanın temel hatlarını oluşturmaktır. GUI için React.js kullanılmıştır.

V. LİTERATÜR TARAMASI

Sıralama Algoritmaları Hakkında Genel Bilgi
Sıralama algoritmaları, veri kümesini belirli bir sıraya göre düzenlemek için kullanılan algoritmalar olarak tanımlanır. Bu algoritmalar, bir dizi veya liste üzerinde çalışır ve elemanları artan veya azalan bir şekilde sıralar. Sıralama algoritmaları, programlama ve veri yapısı alanında yaygın olarak kullanılan temel algoritmalar arasında yer alır.

Sıralama algoritmalarının çeşitli yaklaşımları vardır, örneğin, Bubble Sort, Selection Sort, Insertion Sort, Merge Sort, Quick Sort gibi popüler sıralama algoritmaları bulunmaktadır. Her bir algoritmanın kendine özgü bir çalışma prensibi ve performans karakteristikleri vardır. Bu algoritmaların nasıl çalıştığı ve nasıl analiz edildiği, yazılım geliştiriciler ve öğrenciler için önemli bir konudur.

2.2. Görselleştirme Araçları ve Yaklaşımları

Sıralama algoritmalarını görsel olarak anlatmak ve anlamak için çeşitli araçlar ve yaklaşımlar geliştirilmiştir. Bu araçlar, sıralama algoritmalarının adım adım çalışma süreçlerini görselleştirmek, animasyonlar oluşturmak ve performans analizini sunmak için kullanılır.

Bazı görselleştirme araçları, grafikler, diyagramlar veya animasyonlarla sıralama algoritmalarını açıklayabilir. Bu araçlar, sıralama adımlarını görsel olarak temsil ederek kullanıcıya algoritmanın nasıl işlediğini anlatır. Ayrıca, bazı araçlar algoritmaların karşılaştırma sayılarını, yer değiştirme sayılarını ve zaman karmaşıklığını da gösterir.

Python programlama dili için Tkinter gibi GUI kütüphaneleri, sıralama algoritmalarını görselleştirmek için yaygın olarak kullanılan araçlardan biridir. Tkinter, kullanıcı arayüzü oluşturmak ve grafikleri görüntülemek için kullanılabilir. Ayrıca, Matplotlib gibi grafik kütüphaneleri de sıralama algoritmalarını görselleştirmek için tercih edilen araçlar arasındadır.

Bu literatür taraması, sıralama algoritmaları hakkında genel bilgi ve görselleştirme araçları ile yaklaşımların önemini vurgulamaktadır. Bu proje, mevcut araçları ve yaklaşımları kullanarak sıralama yapılmaktadır.

VI. UYGULAMA SAYFALARI

3.1. Kullanıcı Arayüzü Tasarımı

Projenin kullanıcı arayüzü, Tkinter kütüphanesi kullanılarak tasarlanmıştır. Kullanıcı arayüzü, sıralama algoritmalarını seçme, veri kümesi boyutunu ve özelliklerini ayarlama, görselleştirmeyi başlatma ve duraklatma gibi işlevleri içerir. Arayüz, kullanıcıya rahat bir deneyim sunmak için düzenli ve anlaşılır bir şekilde tasarlanmıştır.

3.2. Tkinter Kütüphanesinin Kullanımı

Tkinter, Python'da kullanıcı arayüzü oluşturmak için yaygın olarak kullanılan bir kütüphanedir. Projede, Tkinter kullanılarak pencere oluşturma, düğmeler, etiketler, kaydırma çubukları, giriş kutuları gibi arayüz bileşenleri oluşturulmuştur. Bu bileşenler, kullanıcının sıralama algoritmalarını seçmesi, veri kümesi özelliklerini ayarlaması ve görselleştirmeyi kontrol etmesi için kullanılır.

3.3. Sıralama Algoritmalarının Uygulanması

Projede, farklı sıralama algoritmaları uygulanmıştır. Bu algoritmalar, Python dilinde yazılmış ve görselleştirme sürecinde adım adım işlenmiştir. Her bir sıralama algoritması, belirli bir veri kümesi üzerinde çalışır ve elemanları sıralar. Algoritmalar, karşılaştırma ve yer değiştirme adımlarını takip ederek verinin sıralanmasını gerçekleştirir.

3.4. Görselleştirme Stratejileri

Projede kullanılan görselleştirme stratejileri, sıralama adımlarını ve verinin değişimini anlaşılır bir şekilde göstermeyi hedefler. Bu stratejiler, veri kümesini grafiksel olarak temsil eden çizim alanına yansıtılır ve her adımda gerçekleşen değişiklikler görsel olarak güncellenir. Ayrıca, algoritmanın karşılaştırma sayılarını, yer değiştirme sayılarını ve çalışma süresini takip etmek için ilgili bilgiler de gösterilir.

Görselleştirme stratejileri, kullanıcıya sıralama algoritmalarının çalışma mantığını anlamak için interaktif bir deneyim sunar. Kullanıcılar, algoritmanın her adımını takip edebilir, verinin nasıl değiştiğini görebilir ve algoritma performansını karşılaştırma sayıları ve yer değiştirme sayılarıyla analiz edebilir. Ayrıca, görselleştirme stratejileri, kullanıcının sıralama algoritmasının performansını farklı veri kümesi boyutları ve özellikleri için değerlendirmesine olanak tanır.

Görselleştirme stratejileri arasında şunlar yer alır:

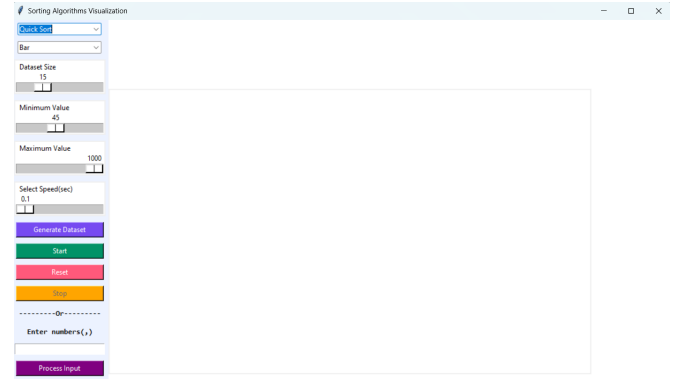
Grafik Temsili: Veri kümesini grafiksel olarak temsil etmek, kullanıcının sıralama işleminin ilerlemesini görsel olarak takip etmesine olanak sağlar. Örneğin, sıralanacak elemanları yatay bir çizgi grafiği olarak göstermek ve her adımda grafiği güncellemek, verinin nasıl sıralandığını anlamayı kolaylaştırır.

Renklendirme: Sıralama adımlarını farklı renklerle vurgulamak, kullanıcının algoritmanın çalışma sürecini daha iyi takip etmesini sağlar. Örneğin, elemanların yer değiştirmesi veya karşılaştırılması durumunda ilgili elemanları farklı renklerle göstermek, değişiklikleri daha belirgin hale getirir.

Animasyon: Sıralama adımlarını animasyonlarla göstermek, kullanıcının algoritmanın çalışma sürecini akıcı bir şekilde izlemesini sağlar. Örneğin, elemanların hareketlerini yavaş yavaş değiştiren bir animasyon oluşturmak, sıralama adımlarını adım adım görselleştirmeyi sağlar.

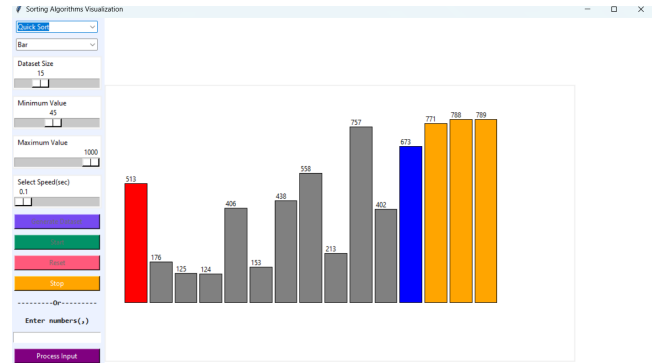
Bu görselleştirme stratejileri, kullanıcıların sıralama algoritmalarını daha iyi anlamalarını ve algoritmaların performansını değerlendirmelerini sağlar. Kullanıcılar, farklı algoritmaları ve veri kümesi boyutlarını deneyerek algoritma analizini gerçekleştirebilir ve sıralama algoritmalarının farklı senaryolarda nasıl performans gösterdiğini gözlemleyebilir.

Bu Tasarım ve İmplementasyon bölümü, projenin kullanıcı arayüzü tasarımını, Tkinter kütüphanesinin kullanımını, sıralama algoritmalarının uygulanmasını ve görselleştirme stratejilerini açıklamaktadır. Bu unsurlar, projenin başarılı bir şekilde gerçekleştirilmesi ve kullanıcılar için interaktif bir sıralama algoritması görselleştirme deneyimi sunması için önemlidir.



Resim- 2

- Uygulamamızda görüldüğü üzere kullanıcıları bu şekilde karşılamaktadır. Sol panelde sıralama algoritmasını ve graf tipini seçebileceği alan mevcuttur Sıralamak istediği sayıları minimum ve maksimum değerlerini random bir şekilde ayarlayabilir veya kendi sayılarını direct yazabilir. Sıralama algoritmasının geçiş hızını ayarlayabilmektedir. Tüm ayarlar yapıldıktan sonra 'Generate Dataset'e basarak uygulama çalışmaya hazır hale gelmiştir. Sonrasında 'Start' tuşuna basarak uygulama çalışmaya başlamaktadır.



Resim- 3



Resim- 4

- Uygulamamızda farklı bir graf tipiyle bir görsel bulunmaktadır. Uygulamayı 'Stop' tuşuyla

durdurabilir ve 'Reset' tuşuyla da sıfırlayabilirsiniz.

REFERENCES

<https://docs.python.org/3/library/tkinter.html>

<https://www.youtube.com/watch?v=IRkvlqPBqNg>

<https://www.geeksforgeeks.org/python-gui-tkinter/>

VII. TARTIŞMA BAŞARILAR VE ZORLUKLAR

Projenin geliştirilme sürecinde birçok başarı elde edildi. İşte bazı başarılar ve zorluklar:

Başarılar:

Kullanıcı Dostu Arayüz: Tkinter kütüphanesinin kullanılmasıyla birlikte kullanıcı dostu bir arayüz tasarlandı. Kullanıcılar, sıralama algoritmalarını kolayca seçebilir, veri kümesi boyutunu ve özelliklerini ayarlayabilir ve görselleştirmeyi kontrol edebilir.

Görselleştirme Stratejileri: Projede kullanılan görselleştirme stratejileri, sıralama algoritmalarının çalışma mantığını ve verinin nasıl değiştiğini anlamayı kolaylaştırdı. Grafik temsili, renklendirme ve animasyon gibi stratejiler, kullanıcıların algoritmanın adımlarını takip etmesini ve performansını değerlendirmesini sağladı.

Algoritma Performansının İncelemesi: Projede kullanıcılar, farklı sıralama algoritmalarını ve veri kümesi boyutlarını deneyerek algoritma performansını inceleyebildi. Bu, kullanıcıların algoritmaların zaman ve bellek karmaşıklığını anlamalarına ve en uygun algoritmayı seçmelerine olanak sağladı.

Zorluklar:

Görselleştirme Süreci: Sıralama algoritmalarını görselleştirmek, bazı zorluklar içeriyordu. Algoritmaların adımlarını anlaşılır bir şekilde göstermek ve verinin değişimini takip etmek için uygun stratejilerin seçilmesi gerekiyordu. Bu, görselleştirme sürecinde bazı zorluklar ve deneme yanılma süreçlerini beraberinde getirdi.

Performans Optimize Etme: Projenin bazı aşamalarında, özellikle büyük veri kümesi boyutlarında performans optimizasyonu gerekiyordu. Algoritmaların verimli bir şekilde çalışması ve görselleştirmenin akıcı bir şekilde gerçekleşmesi için bazı optimizasyon stratejileri uygulanması gerekiyordu.

Geliştirme Önerileri

Projenin geliştirilmesi için aşağıdaki öneriler göz önünde bulundurulabilir:

Farklı Sıralama Algoritmalarının Eklenmesi: Proje, şu anda belirli sıralama algoritmalarını içermektedir. Ancak, farklı algoritmaların eklenmesi kullanıcılar için daha fazla seçenek sunabilir. Örneğin, radix sort, shell sort gibi farklı algoritmaların eklenmesi projeyi daha kapsamlı hale getirebilir.