



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**VERİ YAPILARI ÖDEV RAPORU**

**Verilen Sayıları Belli Bir Düzen İçerisinde Listelere Ayırıp ve  
Listeler Üzerinde Belirli İşlemler Gerçekleştiren Program**

**G211210587- Ekrem Selçuk Çelik**

**SAKARYA**

**Temmuz, 2023**

Veri Yapıları Dersi

**Verilen Sayıları Belli Bir Düzen İçerisinde Listelere Ayırıp ve  
Listeler Üzerinde Belirli İşlemler Gerçekleştiren Program**

Ekrem Selçuk Çelik

*G211210587- 1A- Grubu*

**Özet**

Bir metin dosyasından okunan sayıları iki ayrı bağlı listeye ayırıp, bu listeler üzerinde çeşitli işlemler gerçekleştiren bir C++ programıdır. Her sayının onlar ve birler basamağı ayrı ayrı listelere eklenir. Bu listeler 'top' ve 'bottom' olarak adlandırılır. Projenin ana hedefi, bu listeleri kullanarak belirli bir şekilde sıralama, ortalama hesaplama ve listeleri kullanıcıya yazdırma işlemlerini gerçekleştirmektir. Bu hedeflere ulaşmak için, dosya okuma, bağlı liste yapısı ve ortalama hesaplama gibi C++ dilinin

temel özelliklerini kullanıyoruz. Program, bellek yönetimi konusunda dikkatli bir şekilde tasarlanmış olup, her oluşturulan nesne, kullanımı tamamlandıktan sonra bellekten silinir. Bu sayede, programın çalışması sonucunda herhangi bir bellek sızıntısı oluşmaz.

© 2023 Sakarya Üniversitesi.

Bu rapor benim özgün çalışmamdır. Faydalanmış olduğum kaynakları içerisinde belirttim. Her hangi bir kopya işleminde sorumluluk bana aittir.

C++, Veri Yapıları, Linked List, Index Average

## • GELİŞTİRİLEN YAZILIM

Bu projede, metin dosyasından okunan sayıları işlemek için bağlı liste veri yapısını kullandık. Proje, bağlı listelerin ve temel dosya giriş/çıkış işlemlerinin anlaşılmasını amaçlamaktadır.

Öncelikle, metin dosyasını açıp, her satırdaki sayıları okuduk. Okunan her sayının onlar ve birler basamağı ayrı ayrı listelere ekledik. Her satır için iki ayrı liste oluşturuldu: onlar basamağını içeren 'top' listesi ve birler basamağını içeren 'bottom' listesi. Her sayı okunduğunda, birer yeni düğüm oluşturulup, ilgili listeye ekledik.

Bu işlemler sırasında, düğümleri eklerken sona eklemeyi tercih ettik, çünkü okunan sayıların orijinal sırasını korumak istedik. Swap işlemi, kullanıcıdan alınan iki indeksi (A ve B) kullanarak, A indeksli 'top' listesini B indeksli 'bottom' listesiyle değiştiriyor. Swap işlemi, bağlı listelerin dinamik doğasını ve esnekliğini göstermek açısından iyi bir örnek oluşturuyor.

Bunların yanı sıra, listelerin indexlerinin ortalamasını hesapladık. Beni en çok yoran kısım bu oldu. Üzerinde 2 gün uğraştım neredeyse. İlk önce ödevi anladığımda Liste ortalamasını aldırıyordum ama sonra indexleri yapmam gerektiğini anladım. Epey bir zorladı açıkcası. En sonunda her şeyi yaptıktan sonra çıktı olarak top ve bottom listelerinin ortalamasını gösteriyorum ve program sonlanıyor.

Program sonlandığında, oluşturulan tüm bağlı listeler siliyorum. Programda kullanılan her bellek, programın sonunda serbest bırakılır. Bu şekilde, programın çalıştığı sürece herhangi bir bellek sızıntısı oluşmaz.

Sonuç olarak, bu proje; dosya işlemleri, bağlı listeler, dinamik bellek yönetimi ve temel istatistiksel işlemler gibi çeşitli konuları kapsayan kapsamlı bir uygulamadır. Kullanılan yöntemler, bağlı listelerin esnek yapısını keşfetmemize olanak sağlıyor.

## SONUÇ

Proje üzerinde epey bir uğraştım. Yukarda da bahsettiğim gibi indexlerin ortalamasını alma işlemi beni çok yordu ama bir çok şey öğrendim diyebilirim. Mantığını çok daha iyi kavradım bir çok şeyin. Farklı girdileri denedim ve bir sorun olmadığını gördüm. Teşekkür ederim :)

## Referanslar

- [1] Stackoverflow üzerinde linkedlistler ve index average konusunda bir kaç blogtan yararlandım.
- [2] ChatGPT üzerinde bazı hata mesajları üzerinden yorumlar aldım. Bazı hataları bununla farkedip düzelttim.
- [3] Github üzerinde daha önceden yapılmış index average konusunda bir kaç proje inceledim.

