



T.C.
SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ
VERİ YAPILARI DERSİ ÖDEV RAPORU

3.ÖDEV

B161210044 - Caner KIRCI

SAKARYA
Aralık, 2017
Veri Yapıları Dersi

Caner Kırıcı b161210044 – 1.Öğretim A grubu

Özet

Ödevde bir klasör yolu girip o klasör yolundaki txt dosyalarının okunmasını ve her dosya için önceden oluşturulmuş olan bağıl listeye düğüm eklememizi ve o düğümde .txt dosyasının içindeki kelimeleri tutan ağaç veri yapısı tutmamız isteniyor. Ödevi yaparken ilk olarak ağaçtan başlayıp daha sonra dosyadan okuma ve daha sonra klasörden okuma kısımlarını yaptım. Bu işlemleri yapabildikten sonra klasörden her okuduğu txt için bir düğümde ağaç nesnesi oluşturan Bağıl liste veri yapısını yazdım. En sonda arama ve inorder listeleme işlemleri için işlem sınıfı oluşturdum.

© 2017 Sakarya Üniversitesi.

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

Anahtar Kelimeler: İşlem,Ağac,Bağıl liste,

1. GELİŞTİRİLEN YAZILIM

İlk olarak Ağaç sınıfını tasarlayarak başladım. Denemek için kendi verdiğim kelimeleri ekledim ve çalıştırıp inorder olarak sıraladığına emin olduktan sonra dosyadan veri okuma kısmına geçtim. Dosyadan veri okuma kısmını yaptıktan sonra Bağıl Liste sınıfını tasarladım. Bağıl Liste ve Ağaç sınıflarını tasarlarken sadece ihtiyacım olan metotları yazdım diğer metotlara gereksinim duymadım. Bağıl listeyi de tasarladıktan sonra denemek için klasörden veri okuma kısmını nasıl yapacağımı araştırdım. İnternette dirent.h kütüphanesi kullanılarak bu işlemin yapılacağını ve onun metodunu buldum. Klasörden okuma kısmını da tasarladıktan sonra Bağıl liste kullanmadan denedim. Daha sonra klasörden oku metodunu Bağıl Liste kullanacak şekilde tekrar düzenledim ve İşlem sınıfını tasarlamaya başladım. İlk olarak listele metodunu tasarladım ve en kolayda bu kısım oldu. Önce okuduğu dosya adını ekrana yazdırmayı sonra da sadece ağacın inorder fonksiyonunu çağırmanın yeterli oldu. Bütün kodlar çalıştıktan sonra klasör yolunu kullanıcıdan isteyip ana menüyü tasarlamaya geçtim. Ve ana menüyü de tasarlayıp ödevi bitirdim.

Bana göre ödevin amacı hem problemleri çözme yeteneğimizi geliştirmek hem de hiç bilmediğimiz bir sorunu (klasörden veri okuma) tıpkı bir mühendis gibi kendimiz araştırarak uygulamayı öğretmektir. Tabi son olarak da ağaç Bağıl liste gibi veri yapılarını iyice kavramamız bu ödevin diğer bir amacıydı(bence).

ÇIKTILAR

Klasör yolu hatalı girildiği zaman yine ana menüde Arama yap ve listele seçimleri çıkıyor fakat bunlardan biri seçildiğinde Dosya bulunamadı hatası verip tekrar ana menüye dönüyor ve .txt dosyasının içi boş olduğu durumda ise hiç yeni düğüm oluşturmayıp hata bile vermeden direkt olarak diğer .txt dosyasını okuyor.

Diğer tüm durumlarda program istenildiği gibi gerekiyor. Ancak klasör yolu hatalı girildiğinde programdan çıkıp tekrar girmek gerekiyor. Çünkü klasör yolu kullanıcıdan yalnızca program açıldığında isteniyor.

2. SONUÇ

Bu bölümde çalışmanızdan elde edilen sonuçları, gerçek hayata sağladığı faydalar yazılabilir.

Ağaç veri yapısının nasıl kullanılıp nasıl çalıştığını hangi metotların nasıl yazdığını öğrenmiş olup Bağlı listeyi aktif olarak kullanabilmeyi öğrendim. Birde dirent.h kütüphanesi ile klasörden veri okumayan kod parçasını kullanmayı öğrendim ama kod parçasının nasıl çalıştığını tam olarak anlayamadım.

Bu ödevimin gerçek hayatıma faydası olarak problemlere mühendislik yaklaşımı ile bakmayı ve sorunu çözerken büyük bir problemi küçük parçalara bölerek parçaları birleştirerek çözmeyi öğrenmeme katkısı oldu.

Referanslar

- [1] C/C++ ile Veri Yapıları ve Çözümlü Uygulamalar. Prof. Dr. Nejat Yumuşak, Muhammed Fatih Adak.
- [2] Youtube.com -> Kayhan Ayar -> Veri Yapıları.