



SAKARYA
ÜNİVERSİTESİ

Veri Yapıları 2. Ödev

Ad-Soyad: *Dilara Çetin*

Öğrenci NO: *G221210039*

Öğrenci Mail Adresi: dilara.cetin2@ogr.sakarya.edu.tr

Şube: *2. Öğretim C Grubu*

Dersi Veren: *Dr.Öğr. Üyesi Kayhan Ayar*

İstenenler:

- ağaçlar.txt adlı bir dosya içerisindeki her satırı okuyarak her bir satır için ayrı bir ikili arama ağacı oluşturulmalı.
- Oluşturulan arama ağaçları tek yönlü bağlı listede tutulmalı.
- Oluşturduğumuz bağlı listedeki elemanları, adresleri, düğüm toplamaları, sonraki düğümün adresi olacak şekilde örüntülü bir liste şeklinde ekrana yazdırılmalı. (Sadece on adet.)
- Kullanıcı bu listede “a” ve “d” tuşlarına basarak ileri-geri oklarla hareket edebilmeli.
- Kullanıcı “s” tuşuna bastığında gösterilen düğüm silinmeli.
- Kullanıcı “w” tuşuna bastığında seçili olan düğümün dallarına aynalanma işlemi uygulanmalı.
- Tüm bunları yaparken aynı zamanda seçili olan ağaç level order olacak şekilde ekrana yazdırılmalı.

Yaptıklarım:

İlk olarak BST ağacının düğüm classını (Node) daha sonra BST ağacının classını oluşturdum. Txt dosyasındaki verileri okuyarak oluşturduğum sınıfların doğruluğunu kontrol ettim. Daha sonra bu ağaçları bir bağlı listede tutacak olan sınıfın düğümlerini (BSTListNode) ve kendisini (BSTList) oluşturdum ve txt’yi tekrar okuyarak ağaçları bir bağlı listeye aktardım. Bu aşamada 1. Ödevde edindiğim tecrübeler ve yazdığım sınıflar çok yardımcı oldu.

Daha sonra BSTList classı içerisinde ekrana yazdırma metotlarını oluşturdum. Bu işlem için print, printNode, printNextNode, printSumNodes adlı 4 metot oluşturdum. Bu metotları oluştururken sisteme yüklediğiniz videodan yardım aldım. Daha sonra bir view metodu oluşturarak oluşturduğum print metotlarını burada çağırdım. Ekrana yazdırırken lowerValue ve upperValue adlı iki değişken kullandım. Bu değişkenler bağlı listedeki ağaçların kaçınıcı onluğunda olduğunu kontrol ediyorlar. (Örneğin lowerValue=0 ve upperValue=1 iken 0-10 arası ağaçlar ekrana yazdırılacak). Kullanıcı hareketlerini index adlı bir değişkende tuttum. Bu değişkene göre kullanıcının kaçınıcı ağaçta olduğuna erişebiliyorum ve okları da ona göre yazdırıyorum.

Kullanıcının klavyeden girdiği karakteri usersInput adlı bir metotta kontrol ediyorum. Bu metotta eğer kullanıcı “a” değerini giriyorsa index bir azaltılıyor, “d” değerini giriyorsa index bir artılıyor, “w” değerini giriyorsa mirrortree metoduyla aynalama yapılıyor, “s” değerini giriyorsa seçilen düğüm deleteTree metoduyla siliniyor. Daha sonra seçilen ağaç ekrana yazdırılıyor. Aynı zamanda kalan ağaç sayısına dikkat ederek son ağaç silindikten sonra kullanıcıdan input alırken kodun çökmesini önledim.

Zorlandıklarım ve yapamadıklarım:

Düğüm toplamalarını hesaplamada ve ağaçları ekrana yazdırmada çok zorlandım. Düğüm toplamalarını hesaplarken hala bazı ağaçlarda hatalı hesaplama yapıyorum. Erişemediğim sol düğümler var. Ekrana yazdırma metodunu kendi kuyruk sınıfımı oluşturarak yapmayı denedim fakat yapamadım. Son olarak dizi kullanarak yaptım, her ne kadar ağacı yazdırabilsem de istenilenden uzak bir çıktı elde ettim.

