



SAKARYA ÜNİVERSİTESİ
BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

Veri Yapıları
2. Ödev

Adı: *Mehmet*

Soyadı: *Bosdancı*

No: G221210045

Grup: 2. Öğretim B Grubu

Öğretmen: *Arş.Gör.Dr. HÜSEYİN DEMİRCİ*

Ödev dosyasından bizden Veri.txt içerisindeki sayıları satır satır okuyup her satırdan ayrı bir avl ağacı oluşturulması istendi.

Daha sonra bu avl ağaçlarının yapraklarının bir yığita atılması ve yığıttan sırasıyla bir en küçük bir en büyük elemanın çıkarılması istendi. Aynı ise sayılar ilk oluşan ağaçtan silme işlemi yapıyorduk.

Eğer yığıttaki silme işlemlerinden sonra yığit boşalırsa ilgili ağacın silinmesi istendi ve diğer ağaçlar tekrar karşılaştırma işlemi yaparken ilk haldeki yığıtta ne varsa oradan işlemler gerçekleştirilmeliydi.

Bu şekilde en son kalan ağacın kaç numaralı ağaç olduğu ve ascii değerinin karakter olarak basılması istendi.

Ben avl ağaçları ile yığıtlar arasındaki bağlantıyı avl ağaçlarına "Stack* mystack;" şeklinde gerçekleştirdim.

Bu şekilde avl ağaçları ile stack arasında bağlantı kurabiliyordum.

Ben avl ağaçlarını dizide tutarak işlemlerimi gerçekleştirdim. Bu şekilde indislerle rahatça erişim sağlayabildim.

Ödevde beni en çok zorlayan yerlerden bir tanesi stackler üzerinde yaptığım işlemlerdi.

İlk aşamalarda stackleri de ayrı bir dizide tutuyordum fakat stacklerden bir tanesini silince işlemlerde sorun çıkıyordu.

Bu sorunu stackleri avl ağacının bir özelliği olarak aştım.

Stackten çıkarma işlemi yaparken ilk oluşan ağacın stack'indeki sayı silinmesi gerekiyordu.

Bu sorunu da for döngüsünün içerisinde bulduğu ilk en küçük veya ilk en büyük eleman üzerinde işlemi yaptıktan sonra break ile döngüyü kırarak çözdüm.

Başka bir yaşadığım zorluksa kodun eksi hatalarla dönmesiydi.

Bu sorunun birden fazla sebebi olabilir ama bendeki sebebi en küçük ve en büyük elemanları tuttuğum değişkenin avl ağacı silme işleminden sonra kendisini güncellememesi imiş.

Beni en çok uğraştıran hatalardan birisi de buydu.

Bir en büyük bir en küçük çıkarma işlemi bir adet sayaç kullanarak çözüme ulaştırdım.

Avl ağacım silindikten sonra tekrar yığıtları eski haline getirmek için önce boşalan ağacı sildim daha sonra tüm stackleri boşalttım.

Daha sonra silinen ağaçtaki indise girmeden tekrar yaprakları yığita attım ve işlemlerim bu şekilde devam etti.

Dosyadan okuma işlemlerini getline() fonksiyonu kullanarak gerçekleştirdim.

Önce kendime ağaç sayısını bulmak için dosyanın kaç satır olduğunu buldum

Daha sonra imleci başa alarak ağaçları dizimin içine attım.

Bu işlemi yaparken For döngüsünün değişik bir kullanımını gördüm.

Karakterler bir string içerisinde dönüyordu daha önce hiç kullanmamıştım bunu öğrenmiş oldum.

En son setw() kullanarak son kalan ağacın ascii değerinin karakter karşılığını ve kaçınıcı avl olduğunu yazdırdım.

Bu şekilde program sona ermiş oldu