



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 DEMİRÇİ*

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

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

Eğer yiğittiği silme işlemlerinden sonra yiğit boşalırsa ilgili ağaçın silinmesi istendi ve diğer ağaçlar tekrar karşılaştırma işlemini yaparken ilk haldeki yiğitta ne varsa oradan işlemler gerçekleştirilmeliydi.

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

Ben avl ağaçları ile yiğitler 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 çıktıyordu.

Bu sorunu stackleri avl ağaçının bir özelliği olarak aştım.

Stackten çıkışma işlemi yaparken ilk oluşan ağaçın 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 çözüdüm.

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 ağaçları silme işleminden sonra kendisini güncellememesi imiş.

Beni en çok uğraştıran hatalardan birisi de buydu.

Bir en büyük bir en küçük çıkışma işlemini bir adet sayaç kullanarak çözüme ulaştırdım.

Avl ağaçım silindikten sonra tekrar yiğitleri eski haline getirmek için önce boşalan ağaçtı sildim daha sonra tüm stackleri boşalttım.

Daha sonra silinen ağaçtaki indis girmeden tekrar yaprakları yiğita 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 ağaçın ascii değerinin karakter karşılığını ve kaçinci avl olduğunu yazdırıldım.

Bu şekilde program sona ermiş oldu