

Veri Yapıları 2. Ödev Raporu

Bizden bu ödevde yapılması istenilen bir Veri.txt dosyasında bulunan satırlardaki boşluklarla ayrılmış sayıları her bir satır AVL ağacı oluşturacak şekilde okutmak ve oluşturduğumuz ağacın yapraklarını yığınlarda tutup karşılaştırma işlemi yaparak ağaçlarda yapraklar dışında kalan dallardaki sayıların 26 ile modu alınıp 65 ile toplanması sonucu ASCII harfi bulunup ekrana tüm ağaçlar için yazdırıldıktan sonra yığını biteni yığının ve ağacını silerek en sona kalacak ağacı bulmamızı sağlayacak kodun gerçekleşmesi istendi.

Ben bu ödevde önceki ödev kadar zorlanmadım şahsen. Diğer ödevde nazaran daha hızlı tamamladım ama AVL ağacı mantığı halen karmaşık gelmekte. Bu ödevde başlarken başta tam kavrayamadığımdan ağaç yapısını hocalarımızdan alarak ve biraz araştırarak yazdım. AVL ağacı sınıfının gerçekleşmesi için AVL ağacı düğümlerini sağlayan sınıfı tasarladım. Bu sınıf AVL ağacına ait temel ekleme işlemlerini yapıyor ve onların harf değerlerini buluyor. Sonra bu oluşan ağaçların yapraklarını tutacak bir dinamik dizi ile gerçekleştirdim yığın sınıfı yazdım. Bu sınıf yığına dair temel işlemleri yapmakta. Bu iki sınıfı gerçekleştikten sonra bu ikisinin üzerinde işlemler yapabilecek ve bunları saklayabilecek olan AVL ağaçları sınıfını gerçekleştirdim. “AvlAgaclari” sınıfı silme, ağaçları listeleme, ağaçların yığınlarını saklama, yığınların biri bittikten sonra yeniden dolduracak fonksiyon tanımı ve son harfi bulacak işlemi gerçekleştiren ana fonksiyonumuzu bulunduruyor. lastOneStanding() fonksiyonu bizden ödevde istenilen şeylerin gerçekleştiği en son fonksiyon. Fonksiyonun çalışma mantığı en başta toplam ağaç sayısını veri sayısından alarak indexSilici() ile verilen parametrelerdeki dizilerde tutulan alanları verilen indekse göre siliyor. Ardından yeniStackTanimi() fonksiyonu yığınlardan giden sayıların yenilenmesi için ilk önce var olan tüm yığınlar dizisini siliyor ve kontrol ederekten yığınları indekslerine göre yenileyerek diziye geri yüklüyor. İşlemler yığınlardaki en küçük değeri bul indeksini tut, o yığından küçük değeri çıkar eğer küçük değeri çıkarılan yığının içi boşsa o indeksteki ağacı ve o indeksteki yığını sil ve aynısını büyük değer için de aynı şekilde olacak şekilde yapılmakta, yığınlar yenilenip, kalan ağaç sayısı bir azaltılmakta. Tek fark küçük değer atıldıktan sonra yığın içi boşaltılıp silinirse “continue” kullanılarak döngüye devam edilmeli böylelikle yığın kontrolü bozulmadan istenilen şekilde baştan başlayarak devam edilebilsin. Böyle devam ederek tek ağaç kalana kadar tekrar edecek şekilde lastOneStanding() fonksiyonu sona bir ağaç işaretçisi bırakacaktır. Kalan işaretçiyi diziye yeniden okutarak bulup ekrana yazdıracaktır. Bazı kullanmadığım fonksiyonlar var onlar deneme yanılma yöntemlerim için yazdığım debug fonksiyonlarımdı onları da silmek istemedim.

Bu ödevde en çok zorlandığım kısım kendi mantık hatalarımdı. lastOneStanding() fonksiyonunu yazarken sürekli bir şeyleri eksik bıraktığımdan ötürü her çalıştırdığımda farklı şeylerle karşı karşıya kalıyordum. Bazen program çalışıyor ama bellekte farklı yerlere ulaşmaya çalışıyordu. Dizide sildirme işlemi yaptırdıktan sonra yeniden diziye okurken bellek hataları ile karşı karşıya kalıyordum. Program farklı yerlere erişmeye çalıştığından patlıyordu. Bağlı liste ile gerçeklemeye çalıştığımda da ödev çok yavaş kalıyordu bu da bizden istenilen 40 saniyeyi aşması kısmını bozuyordu. Ben de yeniden diziyle gerçekleştirdim. En sağlıklıısının dizi ile yaptığım olacağını düşündüm. Çünkü dizide tutulan verilerin art arda olması bellekte sürekli atlamalar yapmasından kat kat daha hızlı olmasına yol açıyordu. Hatta bir tür sıralama ya da arama algoritması kullanıp hızlandırmak istedim ama nasıl uygulayacağımı bilemediğimden dizi ile gerçeklemeye geri döndüm. Ödevde eksik bir yer bırakmamaya özen

gösterdim. Elimden geldiğince, anladığım kadarıyla, istenileni sağlamaya çalıştım. Ama ağaçlar üstünde çok fazla işlem yapmadığımdan ötürü ağaçlar hakkında kendimi hala biraz yetersiz hissediyorum. Sadece ağaca ekleme ve ağaç üstünde gezme beraberinde okuma yaptık.