

# VERİ YAPILARI

## 1. Yıl Güz Dönemi 2. Ödev

Enes Bekkaya

B191210034

Bilgisayar Mühendisliği (İ.Ö.) 2.sınıf-A

Dosyadan okunan her üç sayıyı  $x$ ,  $y$  ve  $z$  olarak alıp bir nokta oluşturdum. Oluşturduğum noktaları Nokta sınıfından nesne oluşturarak sakladım ve Nokta sınıfından oluşturduğum nesneyi öncelikli kuyruğa ekledim. Daha sonra öncelikli kuyruk nesnesini Dogru Kuyruğu sınıfından bir nesne oluşturarak Dogru kuyruğu nesnesine ekledim. Son olarak Dogru Kuyruğu nesnesini Avl ağacına ekledim. Bunu dosyadaki her satır için tekrar ettim. Dosyadaki tüm satırları okuduktan sonra avl ağacını postorder olarak dolaştım ve her düğümdeki verinin öncelik sırasını öncelik sırasına göre ekrana çıkardım.

Nokta sınıfında  $x$ ,  $y$ ,  $z$  ve oncelik isminde değişkenler ve öncelikli kuyrukta nesnelerin hangi önceliğe göre saklanacağını belirleyen operatör fonksiyonunu yazdım. Öncelikli kuyruk Node sınıfını kullandığı ve noktaları öncelikli kuyrukta sakladığım için Node sınıfı içinde Nokta türünden bir değişken oluşturdum. Böylece öncelikli kuyrukta nokta sınıfı değişkenlerini saklayabildim. Daha sonra öncelikli kuyruk nesnesini DogruKuyruğu sınıfında saklayabilmek için DogruKuyruğu sınıfı içinde öncelikli kuyruk sınıfı türünde bir pointer ve doğruların uzunluğunu saklamak için toplamUzunluk değişkenini oluşturdum. toplamUzunluk değişkenini avl ağacına yerleştirme yaparken kullandım. Bu pointer öncelikli kuyruk nesnesinin adresini saklıyor. Bu adresle öncelikli kuyruk nesnesinin içinde bulunan noktalar sınıfının nesnesine erişebildim (Bunu postorder dolaşıp ekrana yazdırırken kullandım). Son olarak Avl ağacını oluşturabilmek için AVLnode isimli bir sınıf tasarladım. Bu sınıf içinde DogruKuyruğu tipinde bir pointer oluşturdum ve AVL isminde bir sınıf oluşturarak AVL ağacının bir DogruKuyruğu nesnesini saklayabilmesini sağladım. AVL sınıfı içerisinde gerekli olan fonksiyonları yazdım. Avl ağacına yerleştirmeyi doğruların toplam uzunluğuna göre yaptım ve doğruların uzunluğuna AVLnode da oluşturduğum DogruKuyruğu tipindeki değişken sayesinde DogruKuyruğu nesnesinden eriştim. (DogruKuyruğu sınıfından nesne üretirken bir öncelikli kuyruk ve int tipindeki doğruların toplam uzunluğu verisini yollamamız gerekiyor. Yani doğruların uzunluğunu DogruKuyruğu nesnesinde sakladım.) Ekrana postorder çıkarmak için postorder fonksiyonunu yazdım ve bu fonksiyonda DogruKuyruğu nesnesinden noktalar nesnesine erişebildiğimiz için DogruKuyruğu sınıfında yazdığım yazdır fonksiyonunu çağırdım ve böylece her düğümdeki noktaları öncelik sırasına göre ekrana yazdırdım.

Test cppde verileri dosyadan okudum ve while döngüsü içinde noktaların örijine göre uzaklığını ve satırdaki doğruların toplam uzunluğu hesaplayıp gerekli nesnelere gönderdim. Her satırı okumadan önce new le bir öncelikli kuyruk nesnesi ve her noktayı okumayı bitirdiğimde new le bir nokta nesnesi oluşturdum ve oluşturduğum nokta nesnesini öncelikli kuyruğa ekledim ve böylece satırdaki tüm noktalar aynı öncelikli kuyruk nesnesine eklenmiş oldu. Satırdaki tüm noktaları öncelikli kuyruğa ekleme işlemi bittiğinde öncelikli kuyruk nesnesini her seferinde newle oluşturduğum DogruKuyruğu nesnesine yolladım. Bu işlemi avl ağacının her bir düğümünde tek bir satır tutabilmek için yaptım. Eğer nesneleri newle döngü içinde oluşturmaysaydım avl ağacının düğümünde birden fazla satır tutmuş olacaktım ve bu avl ağacını postorder dolaşıp ekrana öncelik sırasına göre yazdırırken yanlış bir çıktı çıkaracaktı (Döngü içinde newle oluşturmadığım tek nesne avl nesnesidir). Son olarak newle oluşturduğum nesneleri delete komutuyla sildim ve bu sayede çöp oluşmamış oldu.