

Ödev Raporu

Ödevin Konusu:

Bu ödevde verilen bir dosyadaki satırların her biriyle bir ikili arama ağacı oluşturmamız ve bu ağaçları bağlı listeye bağlamamız gerekiyordu. Ayrıca, oluşturulan ağaçların toplam değerini hesaplama, kullanıcı komutlarıyla listeyi ve ağaçları yönetme gibi işlemleri de yapmamız istendi. Dosyadaki karakterlerin ASCII değerleri kullanılarak ağaca yerleştirildiği ve bu değerler üzerinden toplam hesaplandığı bir sistem kurguladım.

Ödevde Yaptıklarım:

İlk olarak, dosyadaki satırları okuyarak her bir satırdan bir ikili arama ağacı oluşturdum. Ağaç yapısını kurarken karakterlerin ASCII değerlerini kullanarak, küçük değerlerin sola, büyük değerlerin sağa yerleştirilmesini sağladım. Daha sonra, bu ağaçları bir bağlı liste yapısına ekledim. Liste üzerinde sayfalama işlemi yaptım ki aynı anda ekranda yalnızca 10 düğüm gösterilebilsin.

Bir diğer önemli nokta ise, ağaçların toplam değerinin hesaplanmasıydı. Sol düğümlerin ASCII değerlerini 2 ile çarparak, sağ düğümlerin değerlerini ise direkt toplayarak toplamı buldum. Kök düğümün de bu toplamın bir parçası olması gerektiğini sonradan fark ettim ve kodu birkaç kez düzeltmek zorunda kaldım.

Ekran çıktıları üzerinde çalışmak da oldukça zaman aldı. Liste ve ağaç yapılarının düzgün çizilmesi için hizalamaları dikkatlice ayarlamam gerekti. Düğümler arasında boşluk bırakma, sayfalama sırasında kaymaların önlenmesi ve seçili düğümün işaretlenmesi gibi detaylar oldukça uğraştırıcıydı.

Karşılaştığım Zorluklar:

Açıkçası, en çok zorlandığım kısım ekran çizimlerini düzgün yapmak oldu. Kodda sürekli kaymalar yaşandı ve özellikle bağlı liste üzerindeki düğümler arasında noktalar koyarken hizalamayı doğru yapmak için çok zaman harcadım. Her düğüm 7 satırlık bir "kutucuk" olarak gösteriliyor ama bu kutuları ekrana doğru yerleştirmek ve hatalı hizalamaları düzeltmek beni bayağı uğraştırdı. Ekrandaki kaymaları çözmek için kodu defalarca yazıp test ettim. Hâlâ çok mükemmel bir çözüm olmayabilir ama en azından istenen çıktıyı almayı başardım.

Bir diğer zorluk ise ağaç toplamı ve aynalama işlemlerinde çıktıların doğru olup olmadığını kontrol etmektir. Özellikle aynalama işlemi sırasında sol ve sağ düğümleri yer değiştirirken, bazen alt ağaçlar birbirine karıştı. Bunun nedeni rekürsif fonksiyonlarda yeterince dikkat etmememdi. Daha sonra alt düğümlerden başlamanın doğru olduğunu fark ettim ve kodu düzelttim.

Sonuç:

Bu ödev benim için hem zorlayıcı hem de öğretici oldu. Daha önce öğrendiğim veri yapılarını (bağlı liste ve ikili arama ağacı gibi) gerçek bir projede kullanmak, bu yapıların dinamik olarak nasıl çalıştığını anlamama yardımcı oldu. Özellikle ekleme, silme ve aynalama işlemlerini yaparken bu yapıların ne kadar esnek olduğunu fark ettim. Hataları bulup çözmek bazen sinir bozucu oldu ama sonuç olarak düzgün çalışan bir sistem elde ettim. Zaten böyle şeyleri uğraştıktan sonra çözmek insana ayrı bir tatmin veriyor.

Bu süreçte hem çok şey öğrendim hem de hata yapmanın öğretici olduğunu bir kez daha gördüm.