



T.C.

SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

VERİ YAPILARI ÖDEV RAPORU

G201210005 - TUĞBA DİRMENCİ

G211210061 - Ash Döngöz

SAKARYA

AĞUSTOS, 2023

VERİ YAPILARI DERSİ

DERS GRUBU:2(A)

1. GELİŞTİRİLEN YAZILIM

1. Kütüphane ve Yapı Tanımlamaları:

- Programın başında, kullanılacak kütüphaneler ve yapılar tanımlanır. Bu kısımda, `'iostream'`, `'fstream'`, `'sstream'` kütüphaneleri giriş/çıkış işlemleri için, `'stack'` kütüphanesi yığıt veri yapısını kullanmak için, `'Node'` ve `'LinkedList'` bağlı liste düğümü ve bağlı liste sınıfını temsil eden yapılar, `'BSTNode'` ve `'BST'` ikili arama ağacı düğümü ve ağacı temsil eden yapılar, `'StackNode'` ve `'StackLinkedList'` yığıt bağlı liste düğümü ve bağlı liste sınıfını temsil eden yapılar, `'BSTNode2'` ve `'BSTLinkedList'` BST bağlı liste düğümü ve bağlı liste sınıfını temsil eden yapılar tanımlanır.

2. `'main'` Fonksiyonu:

- Programın ana fonksiyonu olan `'main'`, dosya işlemlerini gerçekleştirir. - `'ifstream'` ile "Sayilar.txt" dosyası açılır ve dosya okumak için hazırlık yapılır.

3. Dosya Okuma ve Bağlı Liste Oluşturma:

- Dosya başarılı bir şekilde açıldıysa, her satır tek tek okunur.
- Okunan her satır, `'LinkedList'` veri yapısına eklenir. Burada `'LinkedList'` bağlı listesi, satırdaki sayıları tutmak için kullanılır. - `'LinkedList'`'deki her düğüm, satırdaki sayıları bağlı listeye eklemek için kullanılır.

4. Bağlı Listeyi Yığıtlara Bölme:

- `'LinkedList'` içindeki her düğüm, çift sayıları büyükten küçüğe doğru bir araya getirerek `'StackLinkedList'` içinde yığıtlar oluşturur. - Çift sayılar, düğümler arasındaki bağlantıları dikkate alınarak yığıtlara bölünür.

5. Yığıtları BST'lere Dönüştürme:

- `'StackLinkedList'` içindeki her düğüm, içerisindeki sayıları bir `'BST'` oluşturarak `'BSTLinkedList'` içine aktarır.
- Bu işlem sonucunda, her bir düğüm `'BSTLinkedList'` veri yapısında bir ikili arama ağacını temsil eder.

6. BST Ağaçlarını Karşılaştırma:

- `'BSTLinkedList'` içindeki her `'BST'` ağacının yükseklikleri ve düğüm değerleri toplamaları karşılaştırılır.
- En yüksek yüksekliğe sahip ağaç seçilir ve ekrana postorder sırasında düğüm değerleri yazdırılır.
- Bu adımda, yükseklik ve düğüm değeri toplamaları karşılaştırılarak en büyük ağaç seçilir ve postorder sırasında düğüm değerleri ekrana yazdırılır.

7. Dosya Kapatma ve Program Sonlandırma:

- Dosya işlemleri tamamlandıktan sonra, "Sayilar.txt" dosyası kapatılır ve program sonlanır.

3.SONUÇ

Programın amacı, "Sayilar.txt" dosyasında her satırı bir bağlı listeye aktararak, bu bağlı listelerdeki çift sayıları büyükten küçüğe doğru gruplandırıp yığıtlar oluşturmak, daha sonra her yığıtı bir ikili arama ağacına dönüştürmek ve son olarak bu ağaçlardan en yüksek yüksekliğe sahip olanını seçerek postorder sırasında düğüm değerlerini ekrana yazdırmaktır. Bu işlemler, C++ dilindeki bağlı liste, yığın ve ikili arama ağacı kullanımını göstermektedir.