Yöntem

Sağlanan dosyalardaki uygulama, nesne yönelimli programlama prensiplerini kullanır. Her veri yapısı, yapının özel işlemlerini ele alan yöntemlerle kendi sınıfı içinde kapsüllenmiştir.

AVL Ağaçları: AVL ağacı uygulaması, ağacı eklemek, silmek ve dengelemek için yöntemler içerir. Denge, ağacın yükseklik özelliğini korumak için tek veya çift dönüşler aracılığıyla sağlanır.

Yığınlar Yığın, LIFO prensibine bağlı kalarak `push` (ekleme) ve `pop` (çıkarma) gibi temel işlemlerle uygulanmıştır.

AVL ve Yığın Açıklaması

Sağlanan dosyalardaki AVL Ağacı ve Yığın uygulamaları birbiriyle bağlantılıdır. AVL ağacı, belirli işlemler için yığından yararlanır ve bu, farklı veri yapılarının verimli çözümler için nasıl bir arada kullanılabileceğine dair pratik bir örneği gösterir.

AVL Ağaçları: AVL Ağaç düğümleri, sol ve sağ çocuklarına ve bir tamsayı değerine sahip pointerlar ile yapılandırılmıştır. Ağacın dengelemesi kritik bir yönüdür ve her düğümün sol ve sağ alt ağaçları arasındaki yükseklik farkının her zaman bir içinde kalmasını sağlar. Bu denge, kodda etkin bir şekilde uygulanan döndürmelerle sağlanır.

Yığınlar: Yığının uygulanması basit ancak etkilidir ve LIFO kavramını gösterir. Yığın, AVL ağacı bağlamında, derinlik öncelikli gezinmeler gibi işlemler için kullanılır olabilir.

Sonuç

Sağlanan C++ dosyaları aracılığıyla AVL Ağaçları ve Yığınların keşfi, bu veri yapılarının pratik uygulamalarına değerli içgörüler sunar. Kendilerini dengeleyen yapısıyla AVL Ağaçları, verimli arama işlemleri sağlarken; yığınlar, verileri sıralı bir şekilde yönetmek için basit ama güçlü bir yol sunar. Sağlanan kodda bu yapıların birleşimi, bilgisayar bilimlerinde karmaşık sorunları çözmek için farklı veri yapılarının uyum içinde nasıl kullanılabileceğinin pratik uygulamasını ve verimliliğini örnekler.

AVL.cpp:

* AVL Ağacı Sınıfı Oluşturma: Dosya, AVL Ağacı sınıfının bir örneğini oluşturmak için bir constructor (yapıcı) fonksiyon içerir. Bu fonksiyon, AVL ağacının kök düğümünü ve diğer gerekli başlangıç değerlerini ayarlar. Ağacın kök düğümü başlangıçta nullptr olarak ayarlanır, bu da ağacın boş olduğunu gösterir.
* Destructor (Yıkıcı) Fonksiyonu: AVL ağacını silmek için bir destructor fonksiyonu bulunur. Bu fonksiyon, ağacın tüm düğümlerini özyinelemeli bir şekilde silerek ağacın bellekten tamamen temizlenmesini sağlar. Bu, bellek sızıntılarını önlemek için önemlidir.
* Düğüm Silme İşlemi: Ağacın düğümlerini silmek için kullanılan bir yöntem bulunmaktadır. Bu metod, özyineleme (rekürsif) kullanarak ağacın sol ve sağ alt ağaçlarının her bir düğümünü siler. Bu yöntem, destructor içinde kullanılarak ağacın tüm düğümlerinin etkin bir şekilde silinmesini sağlar.
* Ekleme ve Silme Yöntemleri: Dosyada muhtemelen ağaca yeni düğümler eklemek veya mevcut düğümleri silmek için yöntemler bulunmaktadır. Bu yöntemler, AVL Ağacı'nın dengesini korumak için gerekli dönüşler ve ayarlamaları içerir.
* Dengeleme İşlemleri: AVL Ağaçları, her düğümde sol ve sağ alt ağaçlar arasındaki yükseklik farkının bir fazlasını geçmemesini sağlayarak kendilerini dengeleyen ikili arama ağaçlarıdır. Bu dosyada, ağacın dengesini sağlamak için gereken rotasyonlar ve diğer dengeleme işlemleri yer alır.
* Ek Fonksiyonlar ve Yardımcı Yöntemler: Dosya, ağacın yüksekliğini hesaplamak, belirli bir değeri aramak veya ağacın içeriğini belirli bir sırayla listeleyen fonksiyonlar gibi ek işlevler içerir.

AVL\_Tree\_LinkedList.cpp: Bu dosya, AVL ağaç düğümleri için bir bağlı liste uygulamasıdır. Listeden AVL ağaç düğümlerini ekleme ve çıkarma yöntemlerini içerir.

AVLTree\_Node.cpp: Bu dosya, AVL ağaçları için bir düğüm sınıfı tanımlar. Bir sonraki düğümü ayarlama ve almanın yanı sıra AVL ağacı düğümünün değerini alma yöntemlerini içerir.

Node.cpp: Bu dosya, sol ve sağ göstergeler ile bir değer içeren temel bir düğüm yapısını tanımlar, AVL ağacı uygulamasında kullanılır.

Stack.cpp: Bu dosya, bir yığın (stack) veri yapısının uygulamasını içerir. Yığının boş olup olmadığını kontrol etme, yığının üstüne bir değer ekleme ve yığından bir değer çıkarma yöntemlerini barındırır.

StackNode.cpp: Bu dosya, yığın için bir düğüm sınıfı tanımlar. Bir değer ve bir sonraki düğüme işaret eden bir gösterge kullanılmıştır.