**VERİ YAPILARI DERSİ 3. ÖDEVİN RAPORU**

Muzaffer Enes Yapmaz (yapmazenes@gmail.com)

B161210300

Fatih Sözüer (fatihsozuer0@gmail.com)

B161210303

**1.Giriş**

Bizden istenen kısaca yaptığımız uygulama çalıştırıldığında kullanıcıdan bir sayı dizisi alıp bu sayı dizisinin eleman sayısı adedince Ağaç ve soldan sağa doğru okunurken her bir hanenin sayı değerince oluşturulan ağacın yüksekliğinin belirlenmesi ve akabinde bu oluşturulan tüm ağaçların Ana bir ağaçta toplanması sonuç olarak ağaçlardaki En büyük veya En küçük değerlerin çıkartılması ve istenildiğinde ekrana tüm ağaçların Level olarak yazdırılıp çıktı olarak ekrana basılması istenmektedir.

Ödevde yaptıklarımız projede istenilen Ana Ağaç yapısının BST ‘ye göre oluşturulması , çocuk ağaçların oluşturulması , Level Order gezme ve En Büyük ve En Küçük değerlerin bulunup ağaçlardan çıkartılmasıdır.

BST ‘yani Ana ağacın kullanmış olduğu düğümü tasarlarken Veri değişkeninin tipini int yaptık,Böylece rahatlıkla En büyük ve En küçük değere için karşılaştırma yapabilme imkanımız oldu.EnBuyukSil metodu ile ağacın root kökünden o düğümün sağ tarafının gösterdiği pointer NULL ‘u gösterene kadar ağaçta rekürsif bir şekilde dolaştık ve bulmuş olduğumuz son düğümü Ağaçtan sildik.En küçük olanı silme işleminde ise EnKucukSil metodunda root düğümünden başlayarak sol tarafını gösteren pointerın gösterdiği adresin değeri NULL olana kadar rekürsif bir şekilde en sola gittik ve nihai olarak bulduğumuz düğümü BST den Sildik.

AnaAğac’ın kullanmış olduğu düğümü tasarlarken veri kısmını bir BST pointer’ı olarak tasarladık,böylece Ana ağaç üzerinden Heap Bellek bölgesinde oluşturulan BST ağacına erişimi sağlamış olduk.Ana ağacın AraveEkle metodunda ekleme işlemini eklenecek ağacın yüksekliğine göre karşılaştırdık ve ona göre BST mantığında gelen ağacın yüksekliği küçük ise ilgili düğümün soluna şayet büyük ise ilgili düğümün sağına yerleştirdik.Eğer ağaçların yükseklikleri eşit ise ilgili düğümün soluna yerleştirme kabulunu kullandık.Ana ağaçtan silme işlemi yaparken ise ağacın tüm düğümlerini LevelOrder şeklinde gezerek bulunan her bir düğümü bir düğüm pointerı dizisi içerisinde istifledik ve daha sonra doldurulmuş olan dizinin içerisindeki tüm düğümleri En kücük ise çocuk ağacın EnKucukSil metodu ile , En Büyük ise EnBuyukSil metodunu çağırarak silme işlemini gerçekledik.Sonucunda ağacın yüksekliğini güncelledik ve eğer ağacın yüksekliği -1 e eşit ise AnaAğaçtan o düğümü sildik ve heap Bellek bölgesinden ilgili çocuk BST ağacını delete komutu ile temizledik ve çöp oluşma durumunu engelledik.

Ağaçların EnBuyukSil ve EnKucukSil metodlarının çağırılmasında eşit yükseklikli ağaçlardaki yerleşimin bozulma olasılığında ise ağaçların EnBuyuk ve EnKucuk lerini sildikten sonra ana ağacın agacGuncelle() metodunu kullanarak Ana Agacı güncelledik.

AğacGuncelle metodunda ise yeni bir AnaAgac pointer ı tanımlayıp ağacın tüm düğümlerini level order bir şekilde gezme sırasında bulunan her bir düğümün veri kısmındaki BST ‘yi yeni oluşturduğumuz Ana ağacın içerisine ekledik böylece tüm listeler düzgün bir sırada yeniden güncellenmiş oldu ve son olarak eski Ağaç yapısını Temizle() fonksiyonu ile temizledik ve geriye AnaAgacBST ‘si gönderdik ve metodun çağırıldığı yerdeki yani Yonetim sınıfındaki AnaAgac pointerını güncelledik ve elimizdeki AnaAgac bu sayede güncellenmiş oldu.

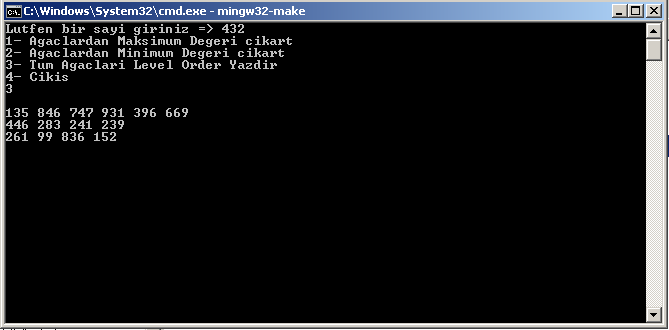
Tüm ağaçların Level Order dolaşılması işlemini ise AnaAğacın içerisindeki düğümleri LevelOrder Mantığı ile dolaşarak ulaşmış olduğumuz her bir Çocuk ağacın içerisindeki Levelorder() metodunu çağırarak ekrana yazdırdık.

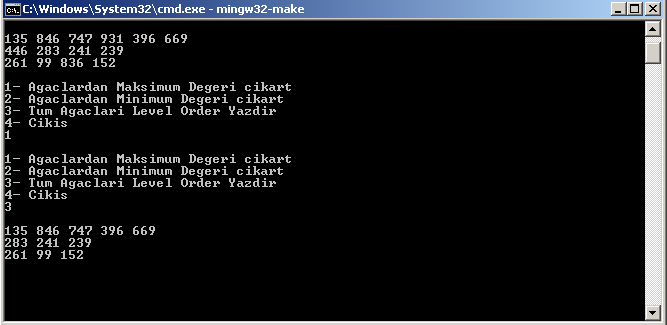
**2-Eksikler-Hatalar-Zorlanılan Kısım**

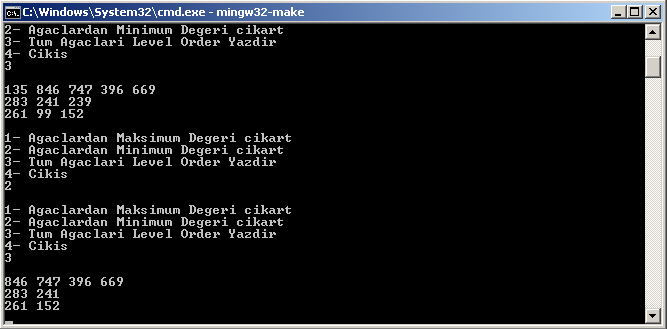
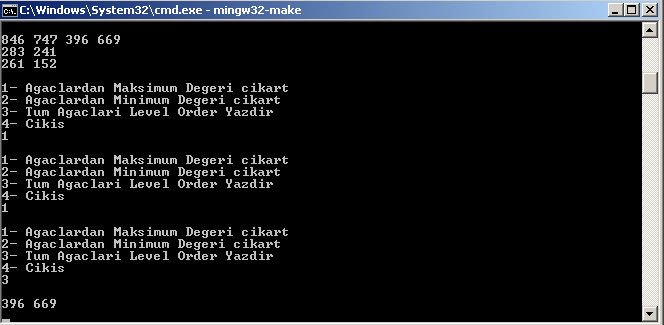
Ödevde herhangi bir eksiklik ve hata bulunmamaktadır.

Zorlanılan kısımlara gelince projenin ilk aşamada nasıl yapılacağına dair fikirlerin zihinde tasarlanması ve ön planın yapılması kısmı ve ek olarak ilk süreçte AnaAgacın nasıl olması gerektiği ve ekleme çıkarma işlemlerinin nasıl yapılması gerektiği kısımlardaki algoritma oluşturma kısmı,ek olarak EnBuyuk ve EnKucuk elemanların silinmesi işleminde direk olarak ulaştığımız noktadaki düğümü silmeye çalıştığımızda AnaAğaçta olası olarak o düğümün tamamiyle elemanlarının bittiğinde AnaAğaçtan silinmesi kısmında Ağacın elemanlarındaki değişiklik sebebiyle bağzı Düğümlerini es geçiyordu ama bu hatanın çözümünü de Nihai olarak elde edilen Düğümleri bir Düğüm pointerı tutan dizide barındırdık ve bu düğüm pointerı dizisinin NULL olmayan hücresindeki Düğümün veri sine ulaşarak gerekli silme işlemlerini gerçekleştirmiş olduk.Burada kullanılan dizi yalnızca Düğümlerin her birini erişebileceğimiz somut bir depoda toplamak ve tüm elemanlarına erişmektir.Yoksa programımızda Ağaç yapımız dizi şeklinde değil normal Liste üzerinde gerçekleştirir şekilde pointerlar üzerinde gerçekleştirilmiştir.

**3.Ödeve ait çıktılar:**

Rastgele Düğüm Ekleme

En Büyük elemanları Çıkar

EnKüçük Elemanları ÇıkarElemanlar bittiğinde Ana Ağaçtan Silinmesi