**Soru** : 1’den 2k-1’e kadar olan sayıları hangi sırada ağaca eklersek tam dengeli bir ikili arama ağacı elde edebilriz?

Dengeli arama ağacı elde edebilmek için önce kök sayı ağaca eklenmeli,daha sonra sol ve sağ çocuklar ağaca eklenmelidir.Bu sebeple fonksiyon her alt ağaç için o alt ağacın kök düğümündeki sayıyı tespit etmektedir(2).Dengeli bir arama ağacında sol alt ağactaki eleman sayısı ile sağ alt ağactaki eleman sayısı aynıdır.o halde kökteki eleman en küçük ve en büyük sayının ortalamasıdır.Fonksiyon daha sonra sırasıyla sol ve sağ alt ağaçlar için kendini çağırmaktadır(5-6).Bu şekilde önce sol alt ağacına ve onunda sol alt ağacına, vs. Eklenecek sayılar belirlenecek,daha sonra fonksiyon geri dönüp sağ alt ağaca eklenecek sayılar belirlenecektir.

1. Void dengeli (int sol,int sag){
2. Int orta=(sol+sag)/2;
3. System.out.print(orta);
4. If(sol!=orta){
5. Dengeli(sol,orta-1);
6. Dengeli(orta+1,sağ);
7. }
8. }

Soru : Verilen bir ikili arama ağacındaki yaprak sayısını hesaplayan fonksiyonu yazınız.

Agac sınıfının içindeki yaprak\_say fonksiyonu bir ağaçtaki yaprak sayısını döndürür.Bunu yapma için Dugum sınıfı içindeki yaprak\_say2 fonksiyonu ağacın kök düğümüyle çağırılır(13).Eğer ağaçta hiç düğüm yoksa fonksiyon 0 döndürür(15).

Dugum sınıfı içindeki yaprak\_say2 ise kök düğümü verilen düğüm olan alt ağaçtaki yaprak sayısını bulur. Bir alt ağaçtaki yaprak sayısı o alt ağacın sol(5-6) ve sağ (7-8) alt ağaçlarındaki yaprak sayılarının toplamına eşittir.(9).Bir alt ağaç sadece yapraktan oluşuyorsa fonksiyon 1 döndürür(3-4).

Int yaprakSay2(){

Int toplam=0;

If(sol==null && sag==null)

Return 1;

If(sol!=null)

Toplam+=sol.yaprakSay2();

If(sag!=null)

Toplam+=sag.yaprakSay2(9;

Return toplam;

}

Int yaprakSay(){

If(kok!=null)

Return kok.yaprakSay2(9;

Else

Return 0;

}

Soru : Verilen bir ikili arama ağacında sayıları seviye seviye ekrana yazan fonksiyonu yazınız.Fonksiyon önce kök düğümü,sonra kök düğümün çocuklarını ,daha sonra kök düğümün çocuklarının çocukalrını vs ekrana yazacaktır.

Ağacın kök düğümünden başlamak üzere(6-10) her düğüm kuyruga eklenir.Her aşamada kuyrugun başından bir düğüm silinir(12-13) ve o düğümün varsa sol (15-18) ve sağ çovukları kuyrugun sonuna eklenir.(16-19).Döngü kuyrukta düğüm kalmayana kadar devam eder(11).Her düğüm için,düğümün çocukları düğümden sonra kuyruga eklendiği için yapılan işlem seviye seviyedir.Önce 1.seviyedeki düğümler sonra 2.seviyedeki düğümler vs . işlenir.

Void seviyeYaz(){

Kuyruk k;

Dugum d;

Eleman e;

K=new Kuyruk(9;

D=kok;

If(d!=null){

E=new Eleman(d);

k.kuyrugaEkle(e);

}

While(!k.kuyrukBos()){

E=k.kuyrukSil();

D=e.dugum;

System.out.print(d.icerik);

If(d.sol!=null){

E=new Eleman(d.sol);

k.kuyrugaEkle(e);

}

If(d.sag!=null){

E=new Eleman(d.sag);

k.kuyrugaEkle(e);

}

}

}

Soru : AVL ağacında durum 2’yi çözmek için uygulanan sol çift rotasyon algoritmasını tek rotasyon fonksiyonlarını kullanmadan yazınız.

AVL ağacında durum 2’yi çözmek için uygulanan sol çift rotasyon algoritması aşağıda verilmiştir.4 tane bağlantı değiştirikmektedir.

\*\*k1’in sağ çocugu artık k2 değiş B diğer bir deyişle k2’nin sol çocugudur.

\*\*k3’ün sol çocugu artık k1 değil C diğer vir deyişle k2’nin sağ çocugudr.

\*\*k2’nin sol çocugu artık B değil k1’dir.

\*\*k2’nin sağ çocugu artık C değil k3’tür.

AvlDugum solCiftRotasyon(AvlDugum k3){

AvlDugum k1,k2;

K1=k3.sol;

K2=k1.sag;

K1.sag=k2.sol;

K3.sol=k2.sag;

K2.sol=k1;

K2.sag=k3;

K1.boy=azami(boy(k1.sol),boy(k1.sag))+1;

K3.boy=azami(boy(k3.sol),boy(k3.sag))+1;

K2.boy=azami(k1.boy,k3.boy)+1;

Return k2;

}

\*\*1. 9,11,7,13,10,5,6,8 sayılarının boş bir ikili arama ağacına eklenmesiyle ortaya çıkan ağacı gösteriniz

Soru : Birinci çözümlü alıştırmada buldugumuz ikii arama ağacından 9’u siliniz.

Soru : Birinci çözümlü alıştırmada buldugumuz ikili arama ağacında önce,ara ve sonra gezinti algoritmalarıyla geziniz.

Soru : Birinci çözümlü alıştırmada buldugumuz ikili arama ağaçlarından hangisi AVL ağacı değildir ve bu ağacı AVL ağacı haline getirmek için hangi rotasyonu uygulamak gerekir gösteriniz.

Soru : Verilen bir ikili arama ağacındaki sayıları büyükten küçüğe ekrana yazan fonksiyonu yazınız.

Soru : Verilen bir ikili arama ağacındaki dolu düğüm sayısını hesaplayan fonksiyonu yazınız.Bir düğümün hem sol hem de sağ çocugu varsa o düğüme dolu düğüm denir.

Soru : Verilen bir ikili arama ağacındaki tek düğüm sayısını hesaplayan foksiyonu yazınız.Bir düğümün değeri tek sayıysa o duğüme tek düğüm denir.

Soru : Verilen bir ikili arama ağacındaki yapraklardan rastgele bir tanesini döndüren fonksiyonu yazınız.

Soru :Verilen bir ikili arama ağacındaki düğümlerden değeri X’ten küçük olanların sayısını döndüren fonksiyonu yazınız.

Soru :Verilen bir ikili arama ağacındaki tüm yaprakları silen fonksiyonu yazınız.

Soru: Verilen bir ikili arama ağacındaki tüm sayıların toplamını bulan fonksiyonu yazınız.

Soru :Verilen bir ağacın ikli arama ağacı koşullarını sağlayıp sağlamadıgını bulan fonksiyonu yazınız.

Soru : Verilen bir ikili arama ağacının tüm düğümlerinin sol ve sağ çocuklarını yer değiştiren fonksiyonu yazınız.

Soru : Verilen bir ikili arama ağacının derinliğini hesaplayan fonksiyonbu yazınız.Bir ağacın derinliği o ağacın kök düğümünden yaprak düğümlerine giden yolların en uzunu olarak tanımlanır.

Soru : Verilen bir ikili arama ağacındaki sayılar k1 ve k2 arasında olanları ekrana yazan fonksiyonu yazınız.

Soru : Verilen bir ikili arama ağacındaki tüm düğümlerin toplam derinliğini hesaplayan fonksiyonu yazınız.

Soru :AVL ağacında durum 3’ü çözmek için uygulanan sağ çift rotasyon algoritmasını tek rotasyon fonksiyonlarını kullanmadan yazınız.