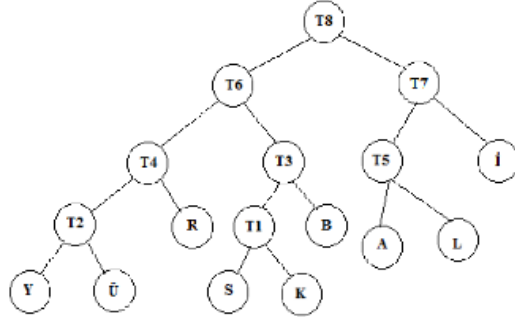


Huffman

- 1) Aşağıdaki huffman kodları ağaca bakarak yazıya çeviriniz. (10 puan)



- a) 010010001011000010001101
SAKARÜL
b) 010100010010110000
KÜRBY
c) 1011101101000001100
LİBSÜA

- 4- Bilgi.txt dosyası incelendiğinde aşağıdaki karakter ve frekanslar ortaya çıkmaktadır. Bu dosyayı Huffman kodlamayı kullanarak Huffman ağacını çiziniz. Her karakterin kodunu bulunuz. Dosyanın durumu ilk durumda ve sıkıştırıldıktan sonra kaç bit olmuştur ayrı ayrı yazınız. Ayrıca dosyada geçen QEZA kelimesi sıkıştırılmadan önce ve sıkıştırıldıktan sonra kaç bit olduğunu belirtiniz (15 p).

Karakter	Frekans
E	32
T	28
A	20
Q	4
Z	1

E: 0

T: 11

A: 101

Q: 1001

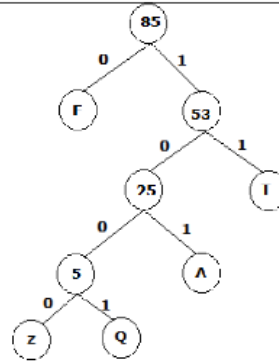
Z: 1000

İlk durumda= $32*8 + 28*8 + 20*8 + 4*8 + 1*8 = 680$

Sıkıştırılmış= $32*1 + 28*2 + 20*3 + 4*4 + 1*4 = 168$

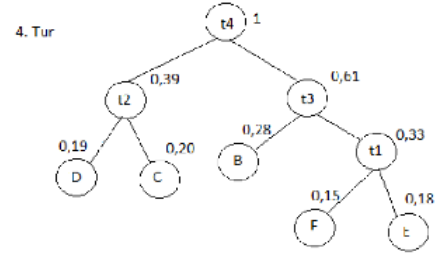
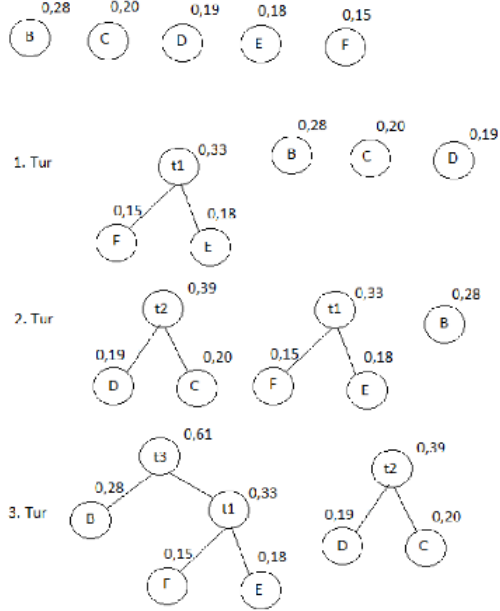
QEZA= $8 + 8 + 8 + 8 = 32$

QEZA(Sıkıştırılmış)= $4 + 1 + 4 + 3 = 12$

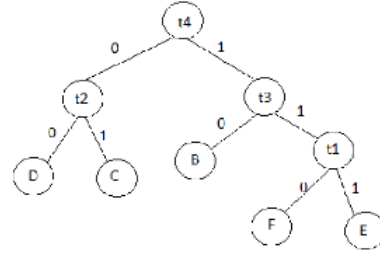


7) Aşağıda bir yazıda karakterlerin **dağılım olasılıkları** verilmiştir. Buna göre Huffman ağacını çizin ve her karakterin yanına kodunu yazınız. (15 p)

Karakter	Olasılık	Huffman Kodu
B	0,28	10
C	0,20	01
D	0,19	00
E	0,18	111
F	0,15	110



Huffman Ağacı



- 9.) a) **yemekleyerekkekelemek** kelimesini Huffman kodlama kullanarak sıkıştırınız. Düğümlerden ağaç oluşturulurken karakter sayısı küçük veya eşit olanlar solda, büyük olanlar ise sağda yer alacak şekilde ağacınızı oluşturunuz. Eşit sayıdaki karakterler için soldan sağa okuyunca ilk gelen harflere öncelik verilmesi gerekmektedir.

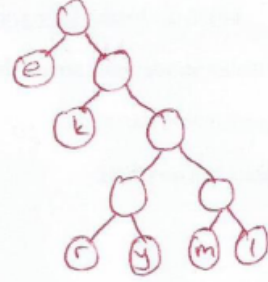
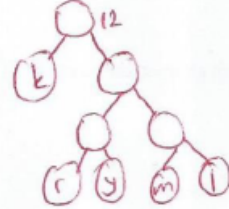
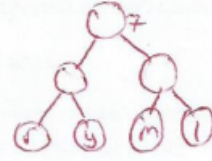
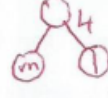
b) Oluşacak ağacınıza göre her bir harfin kodunu yazınız.

c) **ekmek** kelimesinin kodlanmış halini yazınız.

NOT: Çizmiş olduğunuz ağaç doğru olmadıktan sonra yazılan ve doğru olan Huffman koduna puan verilmez. (15 p)

yemeklere kekelemek

-y 2
-e 9
-m 2
-k 5
-l 2
-r 1



y 1101
e 0
m 1110
k 10
l 1111
r 1100

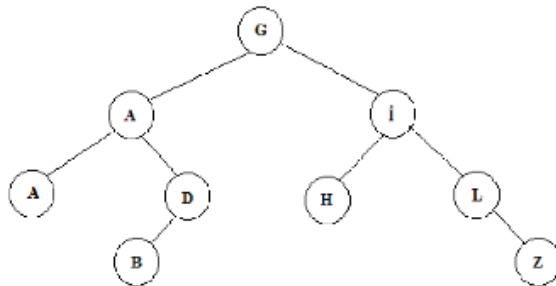
e k m e k
0 10 1110 0 10

5- Yanda frekans tablosu verilen string ifadeye ait Huffman kodlama ağacını oluşturarak, kodlama değerlerini elde ediniz ayrıca bu işleme ait kazanç hesabını yapınız. (15p)

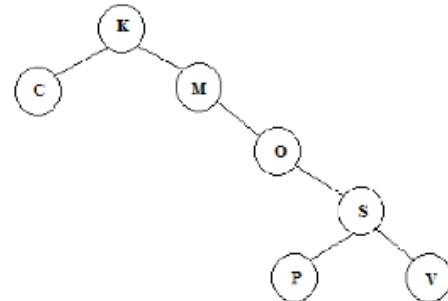
Karakter	Sayı
a	320
b	45
c	60
d	90
e	200
f	70
g	140

Preorder-Postorder

3) Aşağıdaki ağaçları preorder ve postorder olarak okuyup gerekli yere yazınız. (15 puan)

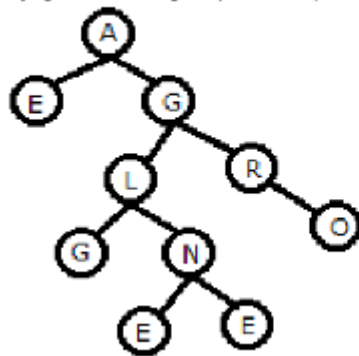


PreOrder : **GAADBİHLZ**
PostOrder: **ABDAHZLİG**



PreOrder : **KCMOSPV**
PostOrder : **CPVSOMK**

2- Aşağıdaki ikili ağacı preorder, inorder ve postorder şeklinde dolaşınız (15 p).



PREORDER: A E G L G N E E R O

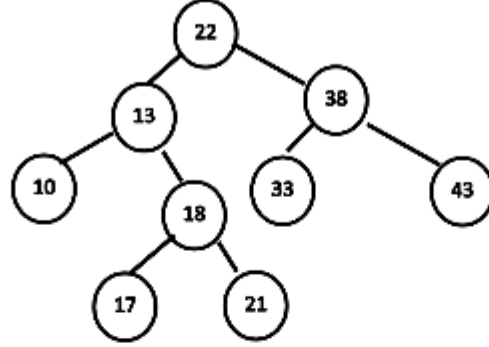
INORDER: E A G L E N E G R O

POSTORDER: E G E E N L O R G A

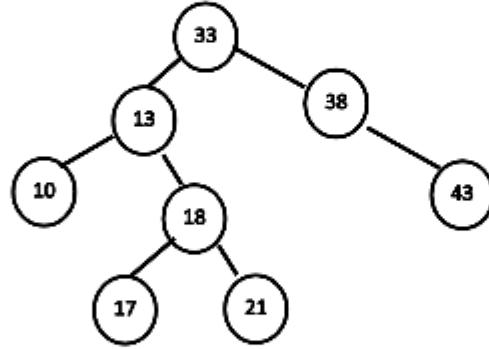
Ağaçlara yerleştirme

5) 22,13,18,38,43,33,17,21,10,22 sayıları sıra ile bir ikili arama ağacına yerleştirilmektedir. **(15 puan)**

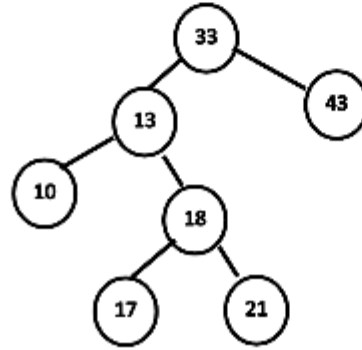
- a) Bu ağacı çiziniz
- b) Ağaçtan 22 sayısı çıkartıldığında ağacın yeni durumunu çiziniz.
- c) Ağaçtan 38 sayısı çıkartıldığında ağacın yeni durumunu çiziniz. (b den sonra yapılacak)



b)



c)



- 8.) a. 77, 22, 9, 68, 18, 34 sayıları ile maksimum heap ağacı oluşturunuz. Her aşamada yapılan işlemler ayrı ayrı gösterilecektir.

b. Oluşturduğunuz bu ağacı kullanarak verilen sayıları büyükten küçüğe sıralayınız.

NOT: HER ADIMDA OLUŞAN AĞAÇ VE YAPILAN İŞLEMİ YAZINIZ. DİREK SONUÇ YAZILDIĞINDA PUAN VERİLMEMEYECTİR.

LIST HEAP YAPISI KULLANILMAMALIDIR (10 p)

AĞAÇ	İŞLEM	AĞAÇ	SIRALAMA LİSTESİ
	77 eklendi		
	22 eklendi		
	9 eklendi		
	68 eklendi		
	22-68 shift up		
	18 eklendi		
	34 eklendi		

2. Sırasıyla 6, 9, 5, 11, 7, 2, 8 sayılarını başlangıçta boş olan ikili arama ağacına (binary search tree) ekleyiniz. Daha sonra sırasıyla 5 ve 9 elemanlarını ağaçtan çıkartınız. (Her bir ekleme ve çıkarma adımını tek tek gösteriniz.)

2. $((a+b-c)*d/e)$ matematiksel ifadesini kullanarak aşağıda istenen işlemleri gerçekleştiriniz.
- Bu ifadeyi ağaç biçiminde ifade ediniz. (5 puan)
 - a şıkında oluşturduğunuz ağaçta preorder dolaşım yaparak oluşan ifadeyi yazınız. (5 puan)
 - a şıkında oluşturduğunuz ağaçta postorder dolaşım yaparak oluşan ifadeyi yazınız. (5 puan)

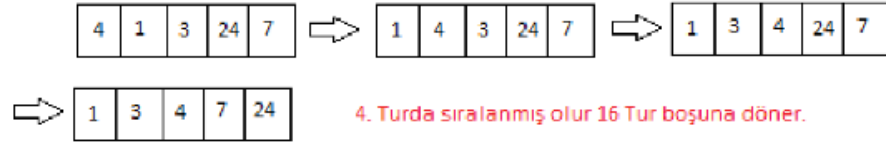
3. Başlangıçta boş olan ikili arama ağacına sırasıyla 54, 36, 49, 65, 18, 76, 71, 58 elemanlarını ekleyiniz (10 puan). Daha sonra sırası ile 65 ve 54 elemanlarını ağaçtan çıkartınız (5 puan). (Her bir işlemi adım adım gösteriniz)

4- "34 54 75 120 110 10 67 70 68 60 130" sayılarını AVL ağacına ekleyip döndür adım adım gösteriniz (15p). Oluşan ağacı postorder gezip yazınız (5p).

Sort

3.) Aşağıdaki diziyi Bubble (Kabarcık) Sort kullanarak sıralayınız. Sıralama işleminin her adımı çizilerek gösterilmelidir. (10p)

4	1	3	24	7
---	---	---	----	---

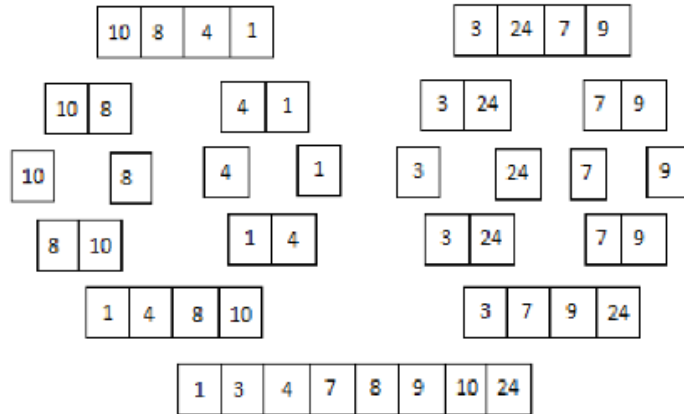


4. Turda sıralanmış olur 16 Tur boşuna döner.

3.) Aşağıdaki diziyi Merge Sort kullanarak sıralayınız. Sıralama işleminin her adımı çizilerek gösterilmelidir. (10p)

10	8	4	1	3	24	7	9
----	---	---	---	---	----	---	---

10	8	4	1	3	24	7	9
----	---	---	---	---	----	---	---



b) 76 93 40 47 10 3 17 66 5

Sayı dizisini 10 uzunluğundaki Hash tablosuna yerleştiriniz. Üç farklı çarpışma giderici yöntem ayrı ayrı kullanılıp ayrı Hash tabloları çizilmelidir. (Çarpışma Giderme: Zincirleme, Açık Adresleme, İkili Hashing) (10 p)

Zincirleme

0		→	10	→	40
1					
2					
3		→	3	→	93
4					
5		→	5		
6		→	66	→	76
7		→	17	→	47
8					
9					

Açık Adresleme

0	40
1	10
2	
3	93
4	3
5	5
6	76
7	47
8	17
9	66

İkili Hashing

0	40
1	3
2	
3	93
4	10
5	17
6	76
7	47
8	66
9	5

R=7

90

4. Bir dizide sırasıyla 8, 2, 7, 1, 6, 9, 4 elemanları bulunmaktadır. Bunları aşağıda istenen sıralama algoritmalarına göre adım adım göstererek sıralayınız.

a. Seçerek Sıralama (Selection Sort) (10 puan)

b. Birleşmeli Sıralama (Merge Sort) (10 puan)

2- {14,6,11,32,7,23,25,13} elemanlarını ikili arama ağacı ile sıralayınız ve dizi yöntemi ile bellekte yerleşimini gösteriniz? (20 p)

Oluşacak Ağaç

Dizi Gösterimi

--	--

Doğrusal sınıama yönteminin kullanıldığı kapalı çarpılama (Closed hashing) için, aşağıdaki elemanları ası ile 7 elemanlı bir tabloya yerleştiriniz. Her bir ekleme işlemini detaylı olarak gösterip tabloyu uşturunuz. (hash: key % 7) (15p)

enecek Değerler: 28, 15, 3, 13, 22, 6

Fibonacci sayılarını hesaplayan Fib fonksiyonunun döngüsel fonksiyon (stack kullanan) hali.

```
int Fib(int sayi)
{
    stack yigin;
    if(sayi==0)
        return 0;

    yigin.push(0);
    yigin.push(1);

    while((--sayi)>0)
    {
        int pop1 = yigin.pop();
        int pop2 = yigin.pop();
        yigin.push(pop1);
        yigin.push(pop1+pop2);
    }
    return yigin.pop();
}
```

Kod yazma

Bir ikili arama ağacındaki en küçük değeri getiren fonksiyonu yazınız. (Yukarıda ağacın ve düğümlerin ön tanımları yapılmıştır. Bütün metotlarının doğru şekilde tanımlandığı varsayılacaktır) (20p)

- Fonksiyonun parametrelerini siz belirleyeceksiniz
- Fonksiyon ayrıca uğradığı her düğümün değerleri toplamını ekrana çıkartmalıdır.
- Fonksiyonun ismi **enKucukGetir** olacaktır.

6- C++ dilindeki `unordered_map` ile `map` arasındaki farkı bir cümle ile yazıp (5p) `map` veri yapısının elemanlarını dolaşacak örnek kodu C++ dilinde yazınız. (10p)

Çerez sorular

7.) a) Dengeli bir ikili arama ağacında yapılacak arama işleminin zaman karmaşıklığı Büyük-O (Big-O) gösteriminde nedir? (5 p)

$O(\log n)$

6-Maksimum düğüm sayısı 255 olan bir binary search tree (BST) için yükseklik kaç olur ? (10 p)

- 7- Bankalardaki müşterileri kabul etme sırası düşünüldüğünde (Kredi kartı ile fiş almak gibi.) en uygun veri yapısı **Öncelikli Kuyruk** verilebilir. (5 p)

Kuyruk yazan 5 puan

Öncelikli Kuyruk yazan 10 puan

BSM207 Veri Yapıları Final Sınavı 11.01.2023

1- (15p)

<pre>class Liste { public: //diğer metotlar //ilk düğüm adresi Dugum* ilk; void islem(); int dugumSayisi; };</pre>	<pre>void Liste::islem() { if(dugumSayisi<2)return; Dugum* gec = ilk; while(gec->sonraki!=NULL) gec=gec->sonraki; ilk->onceki= gec->onceki; gec->onceki->sonraki= ilk; gec->onceki=NULL; gec->sonraki = ilk->sonraki; ilk->sonraki->onceki = gec; ilk->sonraki=NULL; ilk=gec; }</pre>
--	--

Yukarıda iki yönlü bir bağlı liste sınıfının ön tanımı yapılmıştır. islem fonksiyonunun içeriği sağda verilmiştir. Bu fonksiyonun yaptığı işlemi tek bir cümle ile aşağıya belirtiniz.

CEVAP:

6- C++ dilindeki unordered_map ile map arasındaki farkı bir cümle ile yazıp (5p) map veri yapısının elemanlarını dolaşacak örnek kodu C++ dilinde yazınız. (10p)