

Soru : Verilen bir ba lı listenin ikinci pozisyonuna eleman ekleyen fonksiyonu yazınız.

```
Void ikinciyeEkle(Eleman yeni){
If(bas==null)
listeBasinaEkle(yeni);
Else {
If(bas.ileri==null)
listeyeEkle(yeni);
Else {
Yeni.ileri=bas.ileri;
Bas.ileri=yeni; }
}
}
```

Soru : Bir çift ba lı listenin belirli bir parçasını kopyalayıp di er bir listeye yapı tırmak istiyoruz.Kopyalamanın ba ladı ı ve bitti i yer ile kopyalanacak yerin adreslerinin verildi ini varsayarak bu i lemi yapan fonksiyonu yazın.

1. Void kopyalaYapistir(CiftEleman bas,CiftEleman son,CiftEleman kopya){
2. CiftEleman kopyalanan =bas, yapistirilan =kopya;
3. CiftEleman ileri =kopya.ileri,yeni;
4. While(kopyalanan!=son.ileri){
5. Yeni=new CiftEleman(kopyalanan.icerik);
6. Yeni.geri=yapistirilan;
7. Yapistirilan.ileri=yeni;
8. Kopyalanan=kopyalanan.ileri;
9. Yapistirilan=yeni;
10. }
11. Ileri.geri=yapistirilan;
12. Yapistirilan.ileri=ileri;
13. }

Burada **bas** ve **son** deęişkenleri sırayla kopyalamanın başladığı ve bittiği elemanları,**kopya** deęişkeni ise ikinci listede yapıştırılacak pozisyonu göstermektedir.
**her iki listede birer işaretci kullanılarak kopyala-yapıştır işlemi gerçekleştirilmektedir.
kopyalanan listedeki işaretci **kopyalanan deęişkeni ile ,yapıştırılan listedeki işaretci ise **yapistirilan** deęişkeni ile gösterilmektedir.
kopyalanacak her eleman için hafızada yeni eleman için yer açılmakta(5),yeni elemanın geri işaretcisi bir önceki yapıştırılmış elemanı göstermekte(6),bir önceki yapıştırılmış elemanın ileri işaretcisi **yeni elemanı göstermektedir(7).
**Yapıştırılan son elemanın bağlantıları yapıştırılmadan önce kopya dan bir sonra gelen elemene göre yapılmaktadır(11-12).

Soru : Size bir N sayısı veriliyor.Birden N e kadar asal sayıları üreten ve onları bir ba lı liste içinde döndüren fonksiyonu yazınız.

1. Liste asalSayilar(int N){
2. Int i,j;
3. Boolean asal;
4. Liste asallar;
5. Eleman yeni;
6. Asallar=new Liste ();
7. For(i=2;i<N;i++){
8. Asal=true;
9. For(j=2;j<N;j++){
10. If(i%j==0){
11. Asal=false;
12. Break;
13. }
14. }
15. If(asal){
16. Yeni=new Eleman(i);
17. Asallar.listeyeEkle(yeni);
18. }
19. }
20. Return asallar;
21. }

Fonksiyon 2 den N e kadar her i sayısı için asal kontrolü yapmaktadır(7).Bir sayı 1 ve kendisinden başka bir sayıya bölünmüyorsa asaldır.O halde yapılması gereken i nin 2 den N-1 e kadar herhangi bir sayıya bölünüp bölünmediğinin kontrolüdür(10-13).Eğer sayılardan herhangi birine bölünüyorsa i asal olmaz.Aksi takdirde i asal sayılar listesine eklenir(16-17).

Soru : Verilen iki sıralanmış bağılı listenin birleşimini bulan (ortak elemanlar bir kere geçmelidir) fonksiyonu yazınız.

```
Liste birlesim(Liste l1,Liste l2){
    Int icerik;
    Liste sonuc;
    Eleman tmp1,tmp2,yeni;
    Sonuc=new Liste();
    Tmp1=l1.bas;
    Tmp2=l2.bas;
    While(tmp1!=null && tmp2!=null){
        If(tmp1.icerik<tmp2.icerik){
            Icerik=tmp1.icerik;
            Tmp1=tmp1.ileri;
        }else
        If(tmp1.icerik>tmp2.icerik){
            Icerik=tmp2.icerik;
            Tmp2=tmp2.ileri;
        } else {
            Icerik=tmp1.icerik;
            Tmp1=tmp1.ileri;
            Tmp2=tmp2.ileri;
        }
        Yeni = new Eleman(icerik);
        Sonuc.listeyeEkle(yeni);
    }
    Return sonuc;
}
```

Önce sonuc listesi yaratılır.Sonra l1 ve l2 listelerinde sırasıyla tmp1 ve tmp2 işaretçileri kullanılarak aynı anda gezilir.Listeler gezilirken birinci listedeki sayı ikinci listedeki sayıdan küçükse birinci listenin işaretçisi ilerletilir ve birinci listedeki sayı sonuc listesine eklenir.ikinci listedeki sayı birinci listedeki sayıdan küçükse ikinci listenin işaretçisi ilerletilir ve ikinci listedeki sayı sonuc listesine eklenir.her iki sayı aynıysa bir ortak eleman bulunmuş olur , her iki listenin işaretçileri de ilerletilir ve bu ortak eleman yeni sonuc listesine eklenir.