

- 1- Aşağıda verilen fonk1 isimli metot yaptığı işlemi yazınız? Aynı işlemi Yığıt için yapacak metodu C++'ta yazınız. (Ö.Ç. 3, P.Ç. 2, 3, 4)

```
int fonk1(Listeptr *L) {
    int x=0;
    Elemanptr *tmp;
    tmp = L->bas;
    while(tmp != NULL) {
        tmp = tmp->ileri;
        x++;
    }
    return x;
}
```

Cevap: Bağlı listede eleman sayısını bulan fonksiyon

Yığıtın dizi ile gerçekleştirildiğini ve elemanSayisi adında bir değişken tutulduğu varsayılırsa

```
int ElemanSayisi() const {
    return elemanSayisi;
}
```

- 2- Aşağıda verilen fonk1 isimli metot yaptığı işlemi yazınız? Aynı işlemi Dizi için yapacak metodu C++'ta yazınız. (Ö.Ç. 1, 3, P.Ç. 2, 3, 4)

```
Listeptr* fonk1(Listeptr *L1, Listeptr *L2) {
    Listeptr *tmp;
    if(L1->bas == NULL) return L2;
    if(L2->bas == NULL) return L1;
    tmp = new Liste();
    tmp->bas = L1->bas;
    tmp->son = L2->son;
    L1->son->ileri = L2->bas;
    return tmp;
}
```

Cevap: İki bağlı listeyi birleştirme işlemi

```
int* DiziBirlestir(int *d1, int *d2, int uzunluk1, int uzunluk2) {
    int *tmp = new int[uzunluk1 + uzunluk2];
    for(int i=0; i<uzunluk1; i++) {
        tmp[i] = d1[i];
    }
    for(int i=0, j=uzunluk1; j<uzunluk1+uzunluk2; i++, j++) {
        tmp[j] = d2[i];
    }
    return tmp;
}
```

- 3- İki yönlü bağlı listede, listenin başından eleman silmek için C++ kodunu yazınız. Kod olarak sadece ilgili işlem yazılacaktır ve aşağıdaki şekil referans alınacaktır. (Ö.Ç. 3, P.Ç. 2, 3, 4)



Cevap:

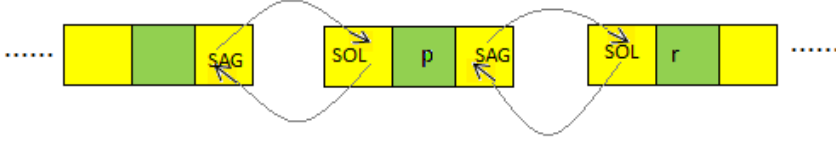
```
x = LB;
LB = LB->sag;
LB->sol = null;
delete x;
return;
```

Ad/Soyad:

Numara:

24.11.2015

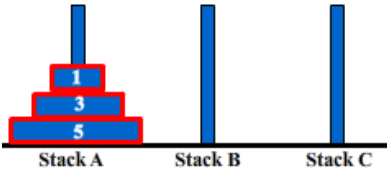
- 4- Aşağıdaki İki yönlü bağlı listede, “p” düğümünün arkasına x elemanını eklemek için C++ kodunu yazınız. Kod olarak sadece ilgili işlem yapılacak ve aşağıdaki şekil referans alınacaktır. (Ö.Ç. 3, P.Ç. 2, 3, 4)



Cevap:

```
void eklesag (dptr p, int *px) {
    dptr q,r;
    if (p==null) {
        cout<<"Geçersiz işlem";
        return;
    }
    q=getDugum();
    q->veri=x;
    r=p->sag;
    r->sol=q;
    q->sag=r;
    q->sol=p;
    p->sag=q;
    return;
}
```

- 5- (Ö.Ç. 5, P.Ç. 1, 2, 3)



Yanda aynı özelliklere sahip üç kule görünmektedir. Bunlardan A kulesi üzerine sıra ile **5 3 1** değerleri yüklenmiştir. Her bir kule bir yığın gibi düşünülürse. Kulelere eleman eklenirken küçük değerler büyük değerler üstüne gelecek şekilde yerleştirilmelidir. (Örneğin 3 değerinin üstüne 5 değeri gelemmez)

Sayıların A yığnında olduğu varsayılacak, sayılar sadece başka bir yığından çıkartılıp diğerine eklenebilir. Yeni bir sayı yığınlara eklenemez, Aşağıda ki programda fonksiyonların yazılı olduğunu varsayarsak bu kurallara göre main fonksiyonunun içeriğini yazınız.

```
class Stack{
public:
    bool Push(int item);
    bool Pop(int& item);
private:
    dizi[10];
    int Top;
}
```

```
int main(){
    Stack A,B,C;
    int temp;
    A.pop(temp);
    C.push(temp)
    A.pop(temp);
    B.push(temp);
    C.pop(temp);
    B.push(temp);
    A.pop(temp);
    C.push(temp);
    B.pop(temp);
    A.push(temp);
    B.pop(temp);
    C.push(temp);
    A.pop(temp);
    C.push(temp);
}
```