

Çift Yönlü Bağlı Liste

Cift_lis.cpp

```
typedef struct Liste
```

```
{
    int numara;
    struct Liste *sonraki;
    struct Liste *onceki;
}BListe;
BListe *ilk=NULL, *son=NULL;
```

```
BListe *Bilgi_Al(int num)
```

```
{
    BListe *bilgi=(BListe *)malloc(sizeof(BListe));
    Bilgi->numara=num;
    Bilgi->sonraki=NULL;
    Bilgi->onceki=NULL;
    return bilgi;
}
```

```
void Ekle(BListe *bilgi)
```

```
{
    if (ilk==NULL)
    {
        ilk=bilgi;
        son=ilk;
    }
    else
    {
        son->sonraki=bilgi;
        bilgi->onceki=son;
        son=bilgi;
    }
}
```

```

void>Listele()
{   BListe *gecici;
    gecici=ilk;
    while(gecici)
    {
        printf("%d\n",gecici->numara);
        gecici=gecici->sonraki;
    }
}

```

```

BListe * Ara(int num)
{   BListe *gecici;
    gecici=ilk;
    while(gecici)
    {
        if(gecici->numara==num) return gecici;
        gecici=gecici->sonraki;
    }
    return NULL;
}

```

```

void Silme(int silnum)
{   BListe *gecici,*bironceki;
    gecici=Ara(silnum);
    if (gecici!=NULL)
    {
        if (ilk==gecici)//silinecek ilk kayitsa
        { if (ilk->sonraki==NULL)//ilk ve tek kayıt var ise
            {

```

```

        ilk=NULL;
        son=NULL;
    }
    else    //ilk kayit ise ancak birden fazla kayit var ise
    {
        ilk=ilk->sonraki;
        ilk->onceki=NULL;
    }
}
else //silinecek kayit ilk kayit degil ise
{
    if(gecici==son)//silinecek son kayit ise
    {son=son->onceki;//son kayit ekleme icin kaybedilmemeli.
    son->sonraki=NULL;
    }
    else //silinecek aradaki kayit arada ise
    {
        gecici->onceki->sonraki=gecici->sonraki;
        gecici->sonraki->onceki=gecici->onceki;
    }
}
    free(gecici);
}
}

```