

Tek Yönü Bağlı Liste Örneği

Genellikle kayıt sayısını bilmediğimiz durumlarda rastgele bir kayıt sayısı vererek dizi boyutunu belirlemek ve gereksiz yere bellekte yer kaplamak yerine, her bir kayıt geldiğinde ona özel dinamik bir bellek alanı açılması, kayıt silineceği durumda da ilgili kayıttan bellekten çıkarılarak bundan sonra gelecek kayıtları için yer açılması statik bir dizi veri yapısı kullanmaktan çok daha etkili olacaktır. Bu durumda kayıtları dinamik olarak oluşturarak aralarında bir bağ varmış gibi kayıtların bir sonraki adresi üzerinden bir birine bağlayarak oluşturulan yapılar bağlı liste veri yapılarıdır. Arada tek yönde bir bağ varsa tek yönlü, çift yönde bir bağ varsa ise çift yönlü bağlı liste yapıları olarak isimlendirilirler.

Örnek: numara bilgisinin tutulacağı tek yönlü bir bağlı liste veri yapısı tanımlayınız. Listeye

- a) Ekleme
- b) Tüm kayıtları Listeleme
- c) Arama
- d) Silme

Fonksiyonlarını yazınız.

Tek List.cpp

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

typedef struct Liste
{
    int numara;
    struct Liste *sonraki;
}BListe;

BListe *ilk=NULL,*son=NULL;

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

void Ekle(BListe *bilgi)
{
    if (ilk==NULL)
    {
        ilk=bilgi;
        son=ilk;
    }
    else
    {
        son->sonraki=bilgi;
        son=bilgi;
    }
}
```

```

}
void main()
{
    BListe *Bilgi,*bul;
    char secim;
    int numara;
    clrscr();
    while(1==1)
    {
        clrscr();
        puts("\nEkleme\nListeleme\nArama\nSilme\nCikis\nSecim?");
        secim=getchar();
        switch(secim)
        {
            case 'e':
                //Bilgi=(BListe *)malloc(sizeof(BListe));
                puts("Numarayi giriniz");
                scanf("%d",&numara);
                Bilgi=Bilgi_Al(num);
                Ekle(Bilgi);
                break;

            case 'a':
                puts("Aranacak Numarayi giriniz");
                scanf("%d",&numara);
                bul=Ara(numara);
                if (bul!=NULL) printf("\nKayit bulundu");else
                printf("\nKayit bulunamadi");
                getch();
                break;

            case 'l':
                Listele();
                getch();
                break;

            case 's':
                puts("Silinecek Numarayi giriniz");
                scanf("%d",&numara);
                Silme(numara);
                break;

            case 'c':exit(0);

        }
    }
}

```