

### 3. Deney Föyü: Tek Yönlü Bağlı Listeler

**Adı Soyadı:**

**Öğrenci No:**

**12.03.2019**

#### 1. Amaç:

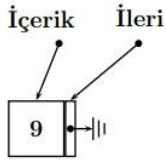
Tek Yönlü bağlı listelerin tanımı, düğüm kavramının incelenmesi, bağlı listeye ekleme, silme ve güncelleme işlemleri

#### 2. Uygulama Dersinden Önce Yapılması Gerekenler

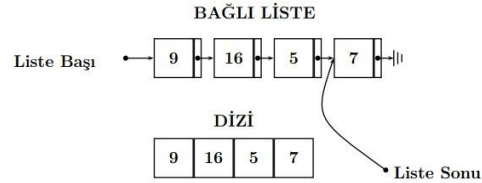
Tek Yönlü bağlı listelerin önerilen kitaplardan incelenmesi gerekmektedir. Bağlı listede bir düğümün nasıl oluşturulduğu araştırılmalıdır. Bir bağlı listede düğümlerin birbirine eklenmesi ve çıkarılması nasıl yapılır? Ayrıca pointer ile düğüm hafızada nasıl tutulur? Bu konuların araştırılması gerekmektedir.

#### 3. Uygulamaya Hazırlık:

Bağlı Listeleri bir küme elemanlarının birbirlerine bir bağlantı bilgisi üzerinden bağlanmış liste yapısıdır. Yalın bir dizide elemanlar birbirini izleyen sırada bulunurlarken, bağlı listede her elemanın veri alanına ek olarak bir bağlantı alanı bulunur ve bu alanlara bir sonraki elemanın adres/indis bilgileri yazılır. Bağlı liste dizi üzerinde tutulsalar dahi bağlantı bilgisi gerekmektedir. Şekil-1’de örnek bir düğüm, Şekil 2’de ise örnek bağlı liste yapısı verilmiştir.



**Şekil 1-** Bağlı liste oluşturmak için örnek bir düğüm yapısı



**Şekil 2 -** Örnek bir bağlı liste yapısı (üstte), standart dizi yapısı (altta)

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

using namespace std;

struct Bliste {
    int veri;
    struct Bliste *ileri;
};

struct Bliste* yeni_bliste = NULL;

void dugum_ekle(int yeni_veri) {
    struct Bliste* yeni_bliste = (struct Bliste*) malloc(sizeof(struct Bliste));
    yeni_bliste->veri = yeni_veri;
    yeni_bliste->ileri = yeni_bliste;
    yeni_bliste = yeni_bliste;
}

void ekrana_yaz() {
    struct Bliste* dugumPtr;
    dugumPtr = yeni_bliste;
    while (dugumPtr != NULL) {
        cout<< dugumPtr->veri <<" ";
        dugumPtr = dugumPtr->ileri;
    }
}
```

#### 4. Örnek Uygulama Kodu:

Örnekte C++ programlama dilindeki struct yapısı ile oluşturulmuş bir düğüm örneği vardır. Bu örnek, bağlı liste yapısına uygun olarak int tipinde tanımlanmış bir tamsayı değeri aracılığıyla **1 adet veriyi** ve bir sonraki düğüm bilgilerini tutmak üzere 1 adet pointer aracılığıyla **bir sonraki düğüme ait indis/adres bilgisini** tutmaktadır.

Ayrıca örnekte, bir adet boş yeni bir düğüm de tanımlanmış olup, bağlı listeye düğüm ekleme ve ekrana yazdırma fonksiyonları da tanımlanmıştır.

Temel olarak C++ programlama dilinde bağlı listeler genellikle **struct yapısı** ve **pointer** kullanılarak yapılır.

```
int main() {
    int key;
    dugum_ekle(5);
    dugum_ekle(12);
    dugum_ekle(3);
    dugum_ekle(23);
    dugum_ekle(4);
    cout<<"Tek Yonlu Bagli Liste: ";
    ekrana_yaz();
    cout<<"\n";
    return 0;
}
```

- Uygulama derslerinde ilgili deney föyünü **mutlaka** yanınızda **bulundurunuz**. **Arkalı önlü çıktı alabilirsiniz**.
- Adınızı, Soyadınızı ve Öğrenci Numaranızı **mutlaka yazınız**.
- Uygulama sorusunun cevabı doğru ise kâğıdın arka sayfasına **yazıp**, kodunuzu kısaca **açıklayınız**.

**5. Uygulama Sorusu .:**

Değerlendiren  
Öğretim Elemanı:

Çalışma  
Durumu:

0

1

2

3

4

5

Not: