

Bağlı Listeler

Bağlı listeler, her bir elemanın bazı değer veya değerler ve bir sonraki elemanın referansını içerdiği veri yapılarıdır. Bir dizi bellekte tüm elemanları için bir blok ayırırken, (bağlı) liste elemanları bellekte ayrı ayrı depolanabilir. Bağlı listelerin her bir elemanına düğüm (node) adı verilir. Bir liste düğümü veri ve referans olarak iki yapı içerir. Liste elemanları birbirlerine adresleri ile bağlıdır, her eleman referans parametresinde bir sonraki elemanın adresini taşır. Listenin ilk elemanı bir işaretçidir, referans değeri sonraki elemanı gösterir. Listenin son elemanı ise diğerlerinden farklı olarak referans değerinde NULL taşır.

Bağlı Listelerin Artıları.

- Bağlı listeler dinamik veri yapılarıdır, yeni eleman eklenebilir ya da eleman silinebilir.
- Etkili bellek kullanımı sağlarlar.
- Eleman ekleme ve çıkarma işlemleri kolaylıkla kodlanabilir, ve hızlı erişim sağlar.

Bağlı Listelerin Eksileri.

- Liste elemanlarına ulaşmak için baştan başlayıp listeyi okumak gerekir.
- Liste elemanlarının adreslerine direk erişim sağlanamaz, çünkü her elemanın adresi bir öncekinden ulaşılır.
- Aynı boyutlarda diziler listeden daha az yer kaplar, çünkü her eleman sadece veriyi taşır.

Ne zaman bağlı liste kullanılır?.

- Ne kadar eleman olacağı bilinmediği durumlarda
- Ekleme/Çıkarma işlemleri için minimum zaman ihtiyacı olduğu zaman
- Herhangi bir elemana rastgele erişim gerekmediğinde
- Listenin herhangi bir yerine eleman eklenmesi gerektiği durumlarda
- Elemanlara erişim zamanı önemli değilse
- Bellek kullanımı önemli ise

Sorular

Site üzerinde verilen arşiv dosyası içindekileri kendi projenize ekleyiniz. Arşiv içinde 3 tane programlama dosyası vardır. Bu dosyalar `linkedlist.h`, `linkedlist.c`, `main.c`.

Bu TP’de kullanacağımız bağlı liste tek yönlü (singly linked) olacak ve sıralı bir şekilde int sayılar depolayacak. Bununla ilgili gerekli struct yapısı `linkedlist.h` içerisinde tanımlanmıştır.

```
1 struct node {
2     int data;
3     struct node* next;
4 }
```

- `linkedlist.c` içerisindeki gerekli fonksiyonları tanımlayınız.
 - `void insert_element(struct node**, int)` Verilen bağlı listeye sıralı (artan şekilde) olarak verilen int değerine sahip düğüm eklemelidir:
 - * Bağlı listede düğüm yoksa başlangıç elemanına (head) yer açılır (malloc) ve int değer atanır.
 - * Bağlı listeye eklenecek int değeri listedeki en küçük değer olacak ise head güncellenir, yani yeni head yeni gelen eleman olur.
 - * Bağlı listeye eklenecek eleman araya eklenmeliyse liste dolaşarak eklenmesi gereken yer bulunur ve bağlantılar güncellenir.
 - * Bağlı listeye eklenecek eleman sona eklenmeliyse liste dolaşıldıktan sonra son elemanın ardına eklenir.
 - `void remove_element(struct node**, int)`
 - * Bağlı liste boş ise herhangi bir işlem gerçekleşmemelidir.
 - * Eğer silinecek eleman head ise yeni head ikinci eleman olacak ve silinecek elemanın memory’deki yeri dinamik olarak boşaltılmalıdır (free).
 - * Eğer listeden çıkarılacak eleman arada veya sonda ise öncesi ve sonrası arasında referans güncellenip silinecek elemanın memory’deki yeri dinamik olarak boşaltılmalıdır (free).
 - `int search_element(struct node* head, int data)` Verilen bağlı listede istenen int değerinde eleman aranır ve bulunan ilk değer kaçınca düğüm olduğu dönülür. Eğer eleman yok ise -1 dönülür.
 - `int get_length(struct node*)` Verilen bağlı listede kaç düğüm olduğu sayılır ve döndürülür. Eğer bağlı liste boş ise 0 döndürülür.
 - `void reverse(struct node**)` Verilen bağlı listenin referans yönü değiştirilir.
 - * Eğer bağlı liste boş veya tek elemanlıysa bir şey yapılmayacak.

- * Bağlı listede birden çok eleman varsa referans tersine döndürülecek. İşlemin sonunda başlangıç durumundaki head'in referansı NULL; yeni head'de listenin sol elemanı olmalıdır (Hint: Bu fonksiyonda üç tane pointer'a ihtiyaç bulunur.)
- Tüm fonksiyonları tanımladıktan sonra `main.c` dosyasında hazırlanmış kod ile deneyiniz, düzgün çalıştığından emin olunuz ve `linkedlist.h`, `linkedlist.c`, `main.c` dosyalarını TP02_Isim_Soyisim.zip formatında Moodle'a yükleyiniz.

```
(base) Timoteos-Macbook:linkedlist macbook$ gcc linkedlist.c main.c -o main.o
(base) Timoteos-Macbook:linkedlist macbook$ ./main.o
Linked list: 1
Linked list: 1 5
Linked list: 0 1 5
Linked list: 0 1 3 5
Linked list: 0 1 3 5 6

Linked list: 0 1 3 5 6

Length of linked list: 5
Index of element 5 in linked list: 3

Reversed linked list
Linked list: 6 5 3 1 0

Linked list: 6 5 3 0
Linked list: 6 3 0
Linked list: 6 3
Linked list: 6
Linked list:
```

Figure 1: Örnek main çıktısı