



Dr. Zeynep Banu ÖZGER



1. Dinamik Bellek
2. Standart Dinamik Bellek Fonksiyonları
3. Malloc Fonksiyonu
4. Calloc Fonksiyonu
5. Realloc Fonksiyonu
6. Free Fonksiyonu



1- DİNAMİK BELLEK

Programın çalışması sırasında;

Bir dizinin uzunluğu değiştirilemez.



Dizinin uzunluğunun çalışma sırasında veya birtakım işlemlerden sonra belirlenmesi gerekebilir.

Bu nedenle dizi uzunlukları genellikle en kötü durum göz önüne alınarak belirlenir.

Belleğin verimsiz kullanılmasına neden olur.



Diziler yerel ise kod bloğu sonlanana kadar, global ise sürekli olarak bellekte tutulur.

Ancak dizi ile işlem bitince ayrılan bellek bölgesinin boşaltılması belleğin verimli kullanılmasını sağlar.



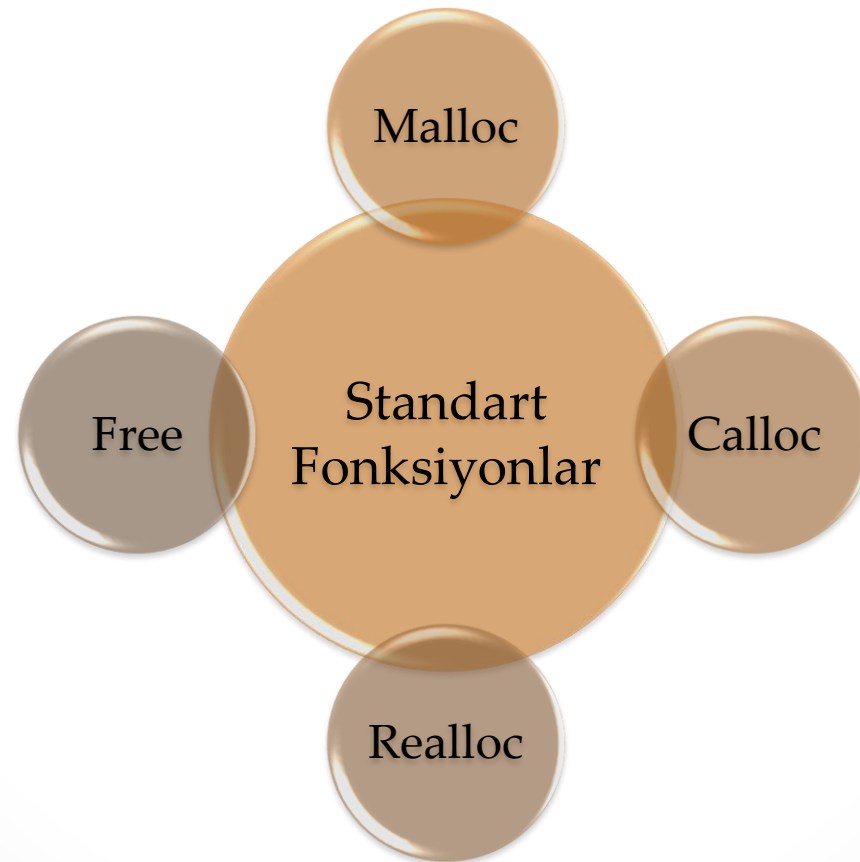
Diziler dinamik bellek fonksiyonları ile kullanılamazlar.



1- DİNAMİK BELLEK

- Dinamik bellek fonksiyonları;
 - Programın çalışması sırasında,
 - Belli uzunlukta bellek bölgesini,
 - Sisteme danışarak tahsis eden ve
 - İstenildiğinde geri bırakan kütüphane fonksiyonlarıdır.







- En yaygın kullanılanıdır.
- `stdlib.h` kütüphanesinde tanımlıdır.
- Parametresi ile belirtilen byte kadar uzunlukta sürekli bellek bölgesi tahsis eder.
- Geri dönüş değeri tahsis ettiği bloğun başlangıç adresidir. **Yani bir pointerdir.**
- Herhangi bir nedenden bellek tahsisi yapamazsa 0 değerine geri döner.
- Genel kullanımı:
 - `void *malloc(size)`



3- MALLOC FONKSİYONU

```
Char *p=malloc(1024)
```

10 byte yer istendi

Tahsis edilen yerin başlangıç adresi p pointerına atandı

Pointer türü bir char

- Sistemde boş bellek alanı yoksa, null değer dönderecektir.
- Tahsisin yapıp yapılmadığının kontrol edilmesi gerekir.
- `if (p!=NULL)`



- Tahsis edilen bölgede bellekte o anda bulunan rastgele değerler vardır.
- Değişken türlerinin bellekte kapladığı alanlar, sisteme göre değişiklik gösterebilmektedir.
- Malloc 'sizeof' ile kullanılarak veri türünün kapladığı alana göre yer tahsis edilmesi sağlanır.

→ 50 elemanlı, int tipinde bir alan istendi



sizeof()

- sizeof () operatörü kendisine verilen değerin bellekte kapladığı alanı verir.
- Verilen değer bir veri türü ise (int, char vs) o türün sistemde kapladığı alanı döndürür.
- Bir değişken ise, değişkenin boyutuna göre sistemde kaç byte yer kapladığını döndürür.
- Geri dönüş değeri bir sayıdır
- Örneğin int veri türü sistemde 4 byte ise
 - sizeof(int)=4
 - int x=5;
sizeof(x)=4
 - int x[5]={1,2,3,4,5};
sizeof(x)=20



SONUÇ?

```
int main()  
{  
    int x=sizeof(int);  
    printf("int turunun sistemde kapladigi alan:%d byte\n",x);  
    int y=5;  
    x=sizeof(y);  
    printf("y degiskeninin sistemde kapladigi alan:%d byte\n",x);  
    int z[5]={1,2,3,4,5};  
    x=sizeof(z);  
    printf("z dizisinin sistemde kapladigi alan:%d byte\n",x);  
}
```



3- MALLOC FONKSİYONU

- Dinamik bellek fonksiyonlarının geri dönüş değeri genel olarak void * 'dır.
- Ancak eski sistemlerde farklı türlerde dönüş yapabilir.
- Bu sorunla karşılaşmamak için geri dönüş değeri belirli bir tipe tür dönüşümü yapılır.
- `p=(int *) malloc (sizeof(int)*100)`

→ Dönüş türü int tipine, tür dönüşümü yapılır



3- MALLOC FONKSİYONU

- Eleman sayısı ve değerleri kullanıcidan alınmak kaydıyla, float tipinde elemanlardan oluşan bir dizinin ortalamasını fonksiyonla bulmak istersek?



4- CALLOC FONKSİYONU

```
void * calloc (count,size)
```

→ count*size kadar bellek tahsis eder

Bellek tahsisatı yaparken
içerisinde malloc
fonksiyonunu çağırır.

Malloc'dan farklı olarak tahsis ettiği bellek bloğunu sıfırlar.

Diğer tüm özellikleri
malloc gibidir.



```
int main()
{
    int n,i;
    char *p1, *p2;
    printf("Dizinin eleman sayisini girin:");
    scanf("%d",&n);
    p1=(char *) malloc (sizeof(char)*n);
    p2=(char *) calloc (n,sizeof(char));
    printf("malloc ile tahsis edilmiş alan\n");
    if (p1!=NULL)
    {
        for(i=0;i<n;i++)
        {
            printf("%d. eleman=%c\n",i+1,* (p1+i));
        }
    }
    printf("\ncalloc ile tahsis edilmiş alan\n");
    if (p2!=NULL)
    {
        for(i=0;i<n;i++)
        {
            printf("%d. eleman=%c\n",i+1,* (p2+i));
        }
    }
}
```



5- REALLOC FONKSİYONU

- Malloc veya Calloc fonksiyonları ile tahsis edilmiş bellek bloklarını büyütmek veya küçültmek için kullanılır.
- stdlib.h içinde tanımlıdır.

```
void * realloc (void * block, newSize)
```



Bloğun toplam yeni uzunluğu

Daha önce tahsis edilen bellek
bloğunun başlangıç adresi

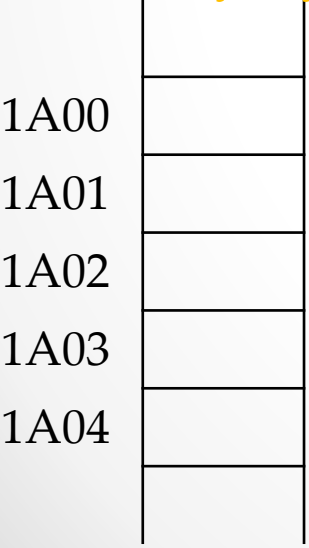


5- REALLOC FONKSİYONU

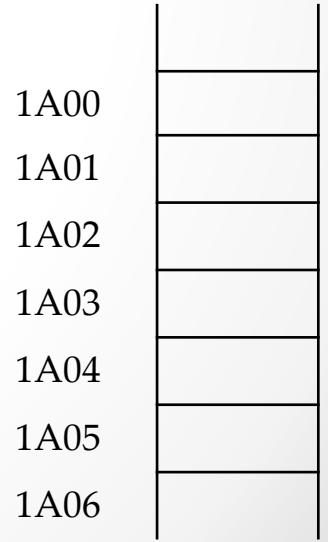
- Realloc, daha önce tahsis edilen bloğun hemen altında, sürekliliği bozmayacak şekilde bellek tahsisi yapar.
- Eğer yeterli yer yoksa bloğun tamamı için bellekte başka yer bakar.
- Dönüş değeri, atanan bloğun başlangıç adresidir.
- Bulamazsa NULL değer dönderir.

```
char *p=(char *) malloc(sizeof(char)*5)
```

→ bellekte 5 byte yer tahsis edildi



```
p=(char *) realloc(p,7)  
→ alanın boyutu 7  
byte'a çıktı
```





5- REALLOC FONKSİYONU

- *Realloc kullanılmadan önce mutlaka malloc veya calloc ile bellek tahsisatı yapılmış olması gerekir.*
- *Tahsis ettiği yeni alan içinde rastgele değerler vardır, tahsis ettiği bloğa ilk değer vermez.*
- *Diziler dinamik bellek fonksiyonları ile kullanılamayacağından, realloc ile dizi boyutu değiştirilemez.*



- Tahsis edilmiş bloklar free ile boşaltılmadıysa, programın sonlanması ile otomatik olarak boşaltılır.



Eleman sayısı ve elemanları kullanıcıdan alınan bir dizinin sonuna, eleman sayısı ve elemanları yine kullanıcıdan alınan 2. bir diziyi eklemek istersek