

STRING DİZİLER (KATARLAR)

Giriş

- ▶ Katar bir char tipinde bildirilen karakter dizisidir.
- ▶ Katarlar (strings), içeriği harfler, rakamlar, veya bazı semboller olan text bilgilerini tutmak(saklamak) için kullanılır.
- ▶ C dilinde string bildirimini için bir tip deyimi yoktur. Bu yüzden, bir stringe bir dizi gözüyle bakılır.
- ▶ Genel olarak bir string'in bildirimini:
`char string_adi[eleman_sayisi];`

`char OgrenciAdi[10];`

Stringlere Başlangıç Değeri Atama

- ▶ Aşağıda verilen iki bildirim aynı anlamdadır:

```
char ktr[5]={'a','h','m','e','t','\0'};
```

```
char ktr[5]="ahmet";
```

- ▶ Eğer bir karakter dizisinin kaç elemandan oluşacağı belirtilmezse, başlangıçta bildirilen karakter sayısı kaç tane ise dizinin eleman sayısının o kadar olduğu varsayılır.

```
char ktr[]="ahmet"; /* 5 elemanlı */
```

Not: `char ktr[];` tanımlaması yanlıştır.

Stringlere Başlangıç Değeri Atama

Stringler bazen bir pointer(işaretçi) ile gösterilebilir.

```
char *ktr="Ankara";
```

Yada

```
char *ad;
```

```
ad="Ankara";
```

Kullanılabilir.

Stringlere Başlangıç Değeri Atama

- ▶ Elemanları string olan diziler tanımlamak mümkündür. Örneğin en uzun 7 karakter olan 5 farklı isim bir çatı altında şöyle toplanabilir:

```
char isim[5][8] = { "Semra", "Mustafa", "Ceyhun", "Asli", "Leyla" };
```

ya da

```
char isim[][8] = { "Semra", "Mustafa", "Ceyhun", "Asli", "Leyla" };
```

ya da

```
char *isim[5] = { "Semra", "Mustafa", "Ceyhun", "Asli", "Leyla" };
```

- ▶ Bu tip tanımlamalarda birinci boyut (sıra) dizinin eleman sayısını, ikinci boyut (sütun) her bir elemanın sahip olabileceği maksimum karakter sayısını gösterir.

Puts ve Gets komutları

- ▶ Bu fonksiyonlar sadece stringler üzerinde işlem yapar. printf(); ve scanf(); fonksiyonları gibi klavyeden veri okuma/ekrana bilgi yazma amaçlı kullanılır.
- ▶ *puts ve gets fonksiyonları*

Örnek-4

```
main(){  
    char ktr[20];  
    printf("\nBir seyler yazın: ");  
    gets(ktr);  
    puts(ktr);  
}
```

Klavyeden girilen bir cümlenin kelimelerini alt alta yazan program.

```
#include<stdio.h>
#include <string.h>
main(){
    char cumle[100];
    int i;
    gets(cumle);
    for (i=0;i<strlen(cumle);i++)
    if (cumle[i]!=' ')
    printf("%c",cumle[i]);
    else
    printf("\n");
    getchar();
}
```

Bazı katar (string) fonksiyonları

`strlen()` Fonksiyonu

Katarın kaç karakterden oluştuğunu hesaplar

Kullanımı : `adet=strlen(str);`

Sonuçta sayısal bir değer döndürür.

strcpy() ve strncpy() ile katar kopyalama

strcpy() fonksiyonu: Bir katarı, bir başka

katara kopyalamak için kullanılır.

Katarlar aynı boyutta olmak zorunda değildir.

Kopyalanacak katar kaynak katardan küçük olsa bile

kopyalama işlemi tam olarak yapılır.

► Kullanımı:

```
strcpy(s1, s2)  
s1=hedef s2=kaynak
```

Örnek:

```
#include<stdio.h>  
#include<string.h>  
main() {  
char kaynak[40]="Merhaba Dünya";  
char kopya[30];  
strcpy( kopya, kaynak );  
printf( "%s\n", kopya );  
}
```

strcpy() ve strncpy() ile katar kopyalama-2

strncpy() fonksiyonu: Strcpy komutuyla aynı görevi görür. Fakat emsalinden farklı olarak, kaç karakterin kopyalanacağı belirtilir.

► **Kullanımı:**

strncpy(s1, s2,sayı)

s1=hedef s2=kaynak sayı=kaynaktan sol baştan kaç karakter alınacak

Örnek:

```
#include<stdio.h>
#include<string.h>
main() {
char kaynak[40]="Merhaba Dünya";
char kopya[30]="";
strncpy( kopya, kaynak ,7);
printf( "%s\n", kopya );
}
```

İlk değer boş da olsa atanmak zorundadır. char kopya[30]; Şeklinde bırakılırsa hatalı sonuç verir.

Ekrana Merhaba kelimesini yazar.

Strcpy örneği

```
▶ main()
{
char isim1[12],isim2[12], baslik[20];

strcpy(isim1,"Ali");
strcpy(isim2,"Veli");
strcpy(baslik,"Bu bir başlıktır.");

printf(" %s ",baslik);
printf("isim 1: %s ",isim1);
printf("isim 2: %s ",isim2);
}
```

strcmp() ve strncmp() ile katar karşılaştırma

strcmp() fonksiyonu, kendisine verilen iki katarı birbiriyle karşılaştırır. Katarlar birbirine eşitse, geriye 0(sıfır) döner. Eğer ilk katar alfabetik olarak ikinciden büyükse, geriye pozitif değer döndürür. Şayet alfabetik sırada ikinci katar birinciden büyükse, geriye negatif değer dönmektedir. Kullanımı:

```
sonuc= strcmp(s1,s2);
```

```
sonuc;
```

-1 ise: s2, s1 den büyük

1 ise: s1,s2 den büyük

0 ise=eşit

dir.

Strcmp komutu örneği

```
#include <string.h>
main(){
    char ktr1 []="Ayhan", ktr2 []="Aydın";
    int sonuc;
    sonuc = strcmp(ktr1,ktr2);
    printf("%d\n",sonuc);
    if(sonuc>0) puts("Birinci ikinciden büyük");
    else if(sonuc<0) puts("İkinci birinciden küçük");
    else puts("eşit");
}
```

Strcmp komutu örneği

```
#include <string.h>
main(){
    char ktr1[10],ktr2[10];
    int sonuc;
    printf("1. katar:");gets(ktr1);
    printf("2. katar:");gets(ktr2);
    sonuc = strcmp(ktr1,ktr2);
    if(sonuc>0) puts(" 1. 2.den büyük");
    else if(sonuc<0) puts("2. 1.den büyük");
    else puts(". eşit");
}
```

strncmp() ile katar karşılaştırma

strncmp() fonksiyonu, kendisine verilen iki katarın başlangıçtan istenilen sayıda karakter birbiriyle karşılaştırır. Katarlar birbirine eşitse, geriye 0(sıfır) döner. Eğer ilk katar alfabetik olarak ikinciden büyükse, geriye pozitif değer döndürür. Şayet alfabetik sırada ikinci katar birinciden büyükse, geriye negatif değer dönmektedir. Kullanımı:

```
sonuc= strncmp(s1,s2,sayi);
```

```
sonuc;
```

```
-1 ise: s2, s1 den büyük
```

```
1 ise: s1,s2 den büyük
```

```
0 ise=eşit
```

```
dir.
```

Sayı: sol baştan itibaren karşılaştırılacak karakter sayısı

Strncmp komutu örneği

```
#include <string.h>
main(){
    char ktr1 []="Ayhan", ktr2 []="Aydın";
    int sonuc;
    sonuc = strncmp(ktr1,ktr2,2);
    printf("%d\n",sonuc);
    if(sonuc>0) puts("Birinci ikinciden büyük");
    else if(sonuc<0) puts("İkinci birinciden büyük");
    else puts("eşit");
}
```


strcat() ve strncat() ile katar birleřtirme

- ▶ *strcat()* ve *strncat()* fonksiyonları, bir katarı bir başka katarla birleřtirmeye yarar. *strcat()* kendisine verilen katarları tamamen birleřtirirken, *strncat()* belirli bir eleman sayısına kadar birleřtirir.

- ▶ **Kullanımları**

strcat(hedef, kaynak);

strncat(hedef, kaynak,sayı);

strcat() Örneği:

```
#include<stdio.h>
#include<string.h>
main()
{
    char ad[30], soyad[20];
    char isim_soyad[50]="";
    printf( "Ad ve soyadınızı giriniz> " );
    scanf( "%s%s", ad, soyad );
    strcat(isim_soyad, ad );
    strcat(isim_soyad," ");
    strcat( isim_soyad, soyad );
    printf( "Tam İsim: %s\n", isim_soyad );
    getchar();
}
```

Eleman sayısı birleştirme yapılacak elemanların sayısından küçük olsa bile işlemi yapar.

Strcat komutu örneği

```
#include <string.h>
main(){
char *a="Ali";
char *b="Veli";
strcat(a,b);
printf(a);
}
```

strncat() Örneği:

```
#include<stdio.h>
#include<string.h>
main()
{
    char ad[30], soyad[20];
    char isim_soyad[50]="";
    printf( "Ad ve soyadınızı giriniz> " );
    scanf( "%s%s", ad, soyad );
    strncat(isim_soyad, ad ,3);
    strcat(isim_soyad," ");
    strncat( isim_soyad, soyad,3 );
    printf( "İsim: %s\n", isim_soyad );
    getchar();
}
```

strstr() fonksiyonuyla katar içi arama yapma

- ▶ *strstr()* fonksiyonu bir katar içinde, bir başka katarı aramak için kullanılır. *strstr()* fonksiyonu, bir katar içinde aranılan bir katarı bulduğu takdirde bulunduğu yerden sonraki katarı pointer değişkene yerleştirir. Eğer herhangi bir eşleşme olmazsa geriye bir sonuç dönmez ve pointer *null* olarak kalır.

- ▶ Kullanımı

Pointer değişken=strstr(str1,str2)

Str1=içinde arama yapılacak katar

Str2=aranacak katar

NOT: Pointer, hafızada başka bir adrese ait değeri tutan, başka bir adresi gösteren değişkene denir.

strstr() Örneği

```
#include<stdio.h>
#include<string.h>
main()
{
    char adres[] = "Anakara";
    char ara[]="kar";
    char *ptr;

    ptr = strstr( adres, ara );
    if( ptr != NULL )
        printf( "Var:%s",ptr );
    else
        printf( "Eşleşme bulunamadı.\n" );

    getchar();
}
```

strchr() ve strrchr() fonksiyonları

- ▶ *strchr()* ve *strrchr()* fonksiyonları, tıpkı *strstr()* gibi arama için kullanılır. Ancak *strstr()* fonksiyonu katar içinde bir başka katarı arayabilirken, *strchr()* ve *strrchr()* fonksiyonları katar içinde tek bir karakter aramak için kullanılır. *strchr()*, karakterin katar içindeki ilk konumunu gösterirken; *strrchr()* fonksiyonu, ilgili karakterin son kez geçtiği adresi verir.

strchr() ve strrchr() fonksiyonları

```
#include<stdio.h>
#include<string.h>
main() {
char adres[] = "Ankara";
char *ilk_nokta, *son_nokta;
ilk_nokta = strchr( adres, 'a' );
son_nokta = strrchr( adres, 'a' );
if( ilk_nokta != NULL ) {
printf( "ilk gorundugu konum: %d\n", ilk_nokta - adres );
printf( "Son gorundugu konum: %d\n", son_nokta - adres ); }
else printf( "Eşleşme bulunamadı.\n" ); }
```


atoi() ve atof() ile katar dönüşümü

- ▶ Verilen katarı, sayıya çevirmek için kullanılır. Eğer metin, bir tam sayıya (*int*) çevrilecekse, *atoi()* , virgüllü sayıya (*float*), çevrilecekse *atof()* fonksiyonu kullanılır. Her iki fonksiyon *stdlib.h* kütüphanesi içindedir. Bu fonksiyonlar kullanılırken, `#include<stdlib.h>` komutu program başlangıcına yazılmalıdır.

atoi() ve atof() ile katar dönüşümü

```
#include<stdio.h>
#include<stdlib.h>
main() {
char kok_iki[] = "1.414213";
char pi[] = "3.14";
char tam_bir_sayi[] = "156";
char yas[] = "18 yaşındayım";
printf( "%d\n", atoi( tam_bir_sayi ) );
printf( "%d\n", atoi( yas ) );
printf( "%f\n", atof( kok_iki ) );
printf( "%f\n", atof( pi ) );
printf("Örnek çarpım:%d",atoi(tam_bir_sayi)*atoi(yas));
getchar();
}
```

Her iki fonksiyon da rakam harici bir şey görene kadar çalışır. Eğer nümerik ifadeler dışında bir karakter çıkarsa, fonksiyon o noktada çalışmayı keser.

strlwr()

Bir dizindeki büyük harfleri küçük harflere çevirir.

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

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

```
main(){
```

```
    char ad[20]="DENEME";
```

```
    printf("%s",strlwr(ad));
```

```
    getch();
```

```
}
```

strupr()

Bir dizindeki harfleri büyük harfe çevirir.

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

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

```
main(){
```

```
    char ad[20]="deneme";
```

```
    printf("%s",strupr(ad));
```

```
    getch();
```

```
}
```

isalnum()

► isalnum(karakter);

Eğer karakter bir harf yada sayı ise fonksiyon değeri sıfırdan farklı; değilse sıfır olur.

```
#include <stdio.h>
#include <conio.h>
main(){
    char harf;
    harf=getchar();
    if (isalnum(harf)!=0)
        printf("Basılan tuş sayı yada karakterdir.");
    else
        printf("Basılan tuş sayı yada karakter değildir.");
    getch();
}
```

isalpha()

- ▶ `int isalpha(karakter);`
- ▶ Eğer `ch` bir harf ise fonksiyon sıfırdan farklı bir değer verir; değilse sıfır verir.

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

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

```
main(){
```

```
    char harf;
```

```
    harf=getchar();
```

```
    if (isalpha(harf)==0)
```

```
        printf("Basılan tuş sayıdır.");
```

```
    else
```

```
        printf("Basılan tuş karakterdir.");
```

```
    getchar();
```

```
}
```

isdigit()

- ▶ `int isdigit(karakter);`
- ▶ Eğer karakter bir sayı ise fonksiyon sıfırdan farklı bir değer verir; değilse sıfır verir.

```
#include <stdio.h>
#include <conio.h>
main(){
    char harf;
    harf=getchar();
    if (isdigit(harf)!=0)
        printf("Basılan tuş sayıdır.");
    else
        printf("Basılan tuş karakterdir.");
    getch();
}
```

Örnek Sorular

- ▶ **Soru 1:** Kendisine verilen bir katarın boyutunu bulan programı yazınız. (Çözüm için `strlen()` fonksiyonunu kullanmayınız.)
- ▶ **Soru 2:** Tersinden de aynı şekilde okunabilen kelime, cümle veya mısraya '*palindrome*' denmektedir. Adı `palindrome()` olan ve verilen katarın tersinin kendisine eşit olduğu durumda geriye 1; aksi hâlde 0 döndüren programı yazınız.

Soru1 Cevabi

```
#include<stdio.h>
#include<string.h>
main( ) {
char katar[50];
int i;
strcpy(katar, "ABCDEF" );
for(i=0; katar[i]!='\0'; i++ );
printf( "Katar boyutu: %d\n",i);
getchar();
}
```

Soru2 Cevabi

```
#include<stdio.h>
#include<string.h>
main() {
char katar[50];
int boyut =0 , i,sonuc;
strcpy( katar, "ABBA" );
//Katar boyutu bulunuyor
for( boyut = 0; katar[ boyut ]!='\0'; boyut++ );
for( i = 0; i < boyut/2; i++ ) {
if( katar[i] != katar[ boyut - i - 1 ] ) sonuc=0;
else sonuc=1 ;
}
printf( "%d\n", sonuc) ;
getchar();
}
```

Ödev

- ▶ Aşağıdaki gibi çalışıp, çıktı üretebilecek programı oluşturunuz.

Merhaba Dünya Nasilsin?

abahreM aynuD ?nislisaN