

Bilgisayar Bilimlerine Giriş-II

-7-

BIL 1002

**Dokuz Eylül Üniversitesi, Fen Fakültesi,
Bilgisayar Bilimleri Bölümü**

Bu günkü dersimizin içeriği

- ▶ Karakter Dizileri (**Strings**) devamı
- ▶ String Kütüphanesinde Karşılaştırma Fonksiyonları
- ▶ String Kütüphanesindeki Arama Fonksiyonları,
<string.h> öncü dosyası
- ▶ String Kütüphanesindeki Hafıza Fonksiyonları
- ▶ String Kütüphanesindeki Diğer Fonksiyonlar
- ▶ Örnekler
 - ▶ **bilgilerini içermektedir**

String Kütüphanesinde Karşılaştırma Fonksiyonları

Prototip	Fonksiyon Tanımı
<code>int strcmp(const char*s1,const char *s2);</code>	s1 stringiyle s2 stringini karşılaştırır. Fonksiyon, s1 s2 'ye eşitse 0 , s1 s2 'den küçükse 0'dan küçük, s1 s2 'den büyükse 0'dan büyük bir değer döndürür.
<code>int strncmp(const char*s1,const char *s2,size_t n);</code>	s1 stringinin n karakterine s2 stringiyle karşılaştırır. Fonksiyon, s1 s2 'ye eşitse 0, s1 s2 'den küçükse 0'dan küçük, s1 s2 'den büyükse 0'dan büyük bir değer döndürür

Bilgisayar bir harfin diğerinden önce geleceğini nasıl bilmektedir? Her karakter, bilgisayarın içinde nümerik kodlar olarak temsil edilir; bilgisayar iki stringi karşılaştırdığında aslında stringler içindeki karakterlerin nümerik kodlarını karşılaştırmaktadır.

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *s1 = "Baris";
    const char *s2 = "Baris";
    const char *s3 = "Ahmet";

    printf("%s%s\n%s%s\n%s%s\n\n%s%d\n%s%2d\n%s%d\n\n",
        "s1 = ", s1, "s2 = ", s2, "s3 = ", s3,
        "strcmp(s1, s2) = ", strcmp( s1, s2 ),
        "strcmp(s1, s3) = ", strcmp( s1, s3 ),
        "strcmp(s3, s1) = ", strcmp( s3, s1 ) );
    printf("%s%2d\n%s%2d\n%s%2d\n",
        "strncmp(s1, s3, 4) = ", strncmp( s1, s3, 4),
        "strncmp(s1, s3, 5) = ", strncmp( s1, s3, 5),
        "strncmp(s3, s1, 5) = ", strncmp( s3, s1, 5) );
    getch();
    return 0;
}
```

```
s1 = Baris
s2 = Baris
s3 = Ahmet

strcmp(s1, s2) = 0
strcmp(s1, s3) = 1
strcmp(s3, s1) = -1

strncmp(s1, s3, 4) = 1
strncmp(s1, s3, 5) = 1
strncmp(s3, s1, 5) = -1
```

String Kütüphanesindeki Arama Fonksiyonları

Not:String kütüphanesindeki fonksiyonları kullanırken,<string.h> öncü dosyasını eklemeyi unutmayın.

Prototip	Fonksiyon Tanımı
char * strchr(const char *s,int c);	c'nin s stringi içindeki ilk konumunu belirler. Eğer c bulunursa, c'yi gösteren bir gösterici döndürülür. Aksi takdirde NULL gösterici döndürülür.
size_t strcspn(const char *s1,const char *s2);	s1 stringinde, s2 stringi içindeki karakterlerden oluşmayan ilk kısmı bulur ve bu kısmın uzunluğunu döndürür.
size_t strspn(const char *s1,const char *s2);	s1 stringi içinde, yalnızca s2 stringi içindeki karakterlerden oluşan ilk kısmı bulur ve bu kısmın uzunluğunu döndürür.
char *strpbrk(const char *s1,const char *s2);	s2 içindeki herhangi bir karakterin, s1 stringi içinde yer aldığı ilk konumu bulur.Eğer s2 stringindeki bir karakter s1 içinde bulunursa,s1 içindeki karakteri gösteren bir gösterici döndürür. Aksi takdirde, NULL gösterici döndürür.

String Kütüphanesindeki Arama Fonksiyonları

Prototip	Fonksiyon Tanımı
char *strrchr(const char *s,int c);	s stringi içinde, c karakterinin en son konumunu belirler. Eğer c bulunursa, s stringi içindeki c 'yi gösteren bir gösterici döndürülür. Aksi takdirde, NULL gösterici döndürülür.
char *strstr(const char *s1,const char *s2);	s2 stringi içinde, s1 dizisinin son konumun belirler. Eğer string bulunursa, s1 stringini gösteren bir gösterici döndürülür. Aksi takdirde, NULL gösterici döndürülür.
char *strtok(const char *s1,const char *s2);	Bir dizi strtok çağrısı s1 stringini,s2 içinde belirtilen karakterle ayrılmış atomlara (bir satırdaki kelimeler gibi mantıklı parçalara) ayırır. İlk çağrı ilk argüman olarak s1 alırken, daha sonraki çağrılar ilk argüman olarak NULL alır. Her çağrıda o andaki atomu gösteren bir gösterici döndürülür. Eğer fonksiyon çağrıldığında daha fazla atom yoksa NULL döndürülür.

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string = "Bu bir testtir";
    char character[2]={ 'a','t'};
    int i;
    for(i=0;i<2;i++)
    {
        if ( strchr( string, character[i] ) != NULL )
            printf( "\"\\%c\\' bu stringde bulundu.\"
                \"\\nHafizadaki konumu: %x \\\"%s\\\".\\n\",
                    character[i], strchr( string, character[i] ),string );
        else
            printf( "\"\\%c\\' bu stringde bulunamadi \\\"%s\\\".\\n\",
                    character[i], string );
    }
    getch();
    return 0;
}

```

```

'a' bu stringde bulunamadi "Bu bir testtir".
't' bu stringde bulundu.
Hafizadaki konumu: 403007 "Bu bir testtir".

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string1 = "Deger 3.14159";
    const char *string2 = "1234567890";

    printf( "%s%s\\n%s%s\\n\\n%s\\n%s%u",
        "string1 = ", string1, "string2 = ", string2,
        "String1'in Strin2'den karakter icermeyen",
        "baslangic kısmi uzunlugu = ",
        strcspn( string1, string2 ) );
    getch();
    return 0;}

```

```

string1 = Deger 3.14159
string2 = 1234567890

String1'in Strin2'den karakter icermeyen
baslangic kısmi uzunlugu = 6

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string1 = "Bu bir testtir";
    const char *string2 = "dikkat";
    printf( "\\\"%s\\\"'%s'%c%s\\n\\\"%s\\\"'%s\\n",
        string2," stringinde gorulen",
        *strpbrk( string1, string2 ),
        " karakteri", string1,"de gorulen "
        "diger stringe ait "
        "ilk karakterdir." );
    getch();
    return 0;
}

```

```

"dikkat" stringinde gorulen'i karakteri
"Bu bir testtir"de gorulen diger stringe ait ilk karakterdir.

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string1 = "Bir hayvanatbahcesi
zebrayida""iceren pek cok hayvana sahiptir";
    int c = 'p';
    printf( "%s\\n'%c'%s\\\"%s\\\"\\n",
        "string1'in son gorulen ",c,
        " karakterinden sonra geriye kalan kismi"
        , strchr( string1, c ) );
    getch();
    return 0;
}

```

```

string1'in son gorulen
'p' karakterinden sonra geriye kalan kismi"ptir"

```



```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string1 = "abcdefabcdef";
    const char *string2 = "def";
    printf( "%s%s\n%s%s\n\n%s%s\n",
            "string1 = ", string1, "string2 = ", string2,
            "string1 icinde string2 karakterleri\n"
            "ile karsilasildigindan itibaren"
            " string1:",
            strstr( string1, string2 ) );
    getch();
    return 0;
}

```

```

string1 = abcdefabcdef
string2 = def

string1 icinde string2 karakterleri
ile karsilasildigindan itibaren string1:defabcdef

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char string[] = "Bu cumle 5 kelimeden olusur.";
    char *tokenPtr;
    printf( "%s\n%s\n\n%s\n",
            "Kelimelerine ayirilacak string:", string,
            "Kelimeler:" );
    tokenPtr = strtok( string, " " );
    while ( tokenPtr != NULL ) {
        printf( "%s\n", tokenPtr );
        tokenPtr = strtok( NULL, " " );
    }
    getch();
    return 0;
}

```

```

Kelimelerine ayirilacak string:
Bu cumle 5 kelimeden olusur.

Kelimeler:
Bu
cumle
5
kelimeden
olusur.

```

String Kütüphanesindeki Hafıza Fonksiyonları

Prototip	Fonksiyon Tanımı
void * memcpy(void *s1,const void *s2,size_t n);	s2 ile gösterilen nesneden n karakteri s1 ile gösterilen nesneye kopyalar. Sonuçta, oluşan nesneyi gösteren bir gösterici döndürülür.
void *memmove(void *s1,const void *s2,size_t n);	s2 ile gösterilen nesneden n karakteri s1 ile gösterilen nesneye kopyalar. Kopyalama işlemi, s2 ile gösterilen nesnedeki karakterler önce geçici bir diziye kopyalanıp daha sonra da bu geçici diziden s1 ile gösterilen nesneye kopyalanıyormuş gibi yapılır. Sonuçta, oluşan nesneyi gösteren bir gösterici döndürülür.
int memcmp(const void *s1,const void *s2,size_t n);	s1 ve s2 ile gösterilen nesnelerin ilk n karakterlerini karşılatır. Fonksiyon, s1 s2 'ye eşitse 0, s1 s2 'den küçükse 0'dan küçük, s1 s2 'den büyükse 0'dan büyük bir değer döndürür.

String Kütüphanesindeki Hafıza Fonksiyonları

Prototip	Fonksiyon Tanımı
void *memchr(void *s,int c,size_t n);	s ile gösterilen nesne içinde c 'in (unsigned char'a dönüştürülür) ilk bulunduğu konumu belirler. Eğer c bulunursa, nesne içindeki c 'in konumunu gösteren bir gösterici döndürülür. Aksi takdirde, NULL döndürülür.
void *memset(void *s,int c,size_t n);	s ile gösterilen nesnenin ilk n karakterine, c(unsigned int'e dönüştürülür) kopyalar. Sonucu gösteren bir gösterici döndürülür.

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char s1[ 17 ], s2[] = "Stringi kopyala";

    memcpy( s1, s2, 17 );
    printf( "%s\n%s\n%s\n",
        "s2,s1'e kopyalandiktan sonra",
        "s1 = ", s1 );
    getch();
    return 0;
}

```

```

s2,s1'e kopyalandiktan sonra
s1 = "Stringi kopyala"

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char x[] = "Evim evim guzel evim";

    printf( "%s%s\n",
        "memmove'dan once x: ", x );
    printf( "%s%s\n",
        "memmove'dan sonra x: ",
        memmove( x, &x[ 5 ], 15 ) );
    getch();
    return 0;
}

```

```

memmove'dan once x: Evim evim guzel evim
memmove'dan sonra x: evim guzel evim evim

```

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *s = "String";
    printf( "%s\\%c\\%s\\\"%s\\\"\\n",
            "String'in ", 'r',
            " karakterinden sonra kalani"
            , memchr( s, 'r', 16 ) );
    getch();
    return 0;
}

```

String'in 'r' karakterinden sonra kalani"ring"

```

#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char string1[ 15 ] = "BBBBBBBBBBBBBBBB";

    printf( "string1 = %s\\n", string1 );
    printf( "memset'ten sonra string1 = %s\\n",
            memset( string1, 'b', 7 ) );
    getch();
    return 0;
}

```

string1 = BBBBBBBBBBBBBBBB
memset'ten sonra string1 = bbbbbbbBBBBBBBB

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    char s1[] = "ABCDEFGFG", s2[] = "ABCDXYZ";

    printf( "%s%s\n%s%s\n\n%s%2d\n%s%2d\n%s%2d\n",
            "s1 = ", s1, "s2 = ", s2,
            "memcmp( s1, s2, 4 ) = ", memcmp( s1, s2, 4 ),
            "memcmp( s1, s2, 7 ) = ", memcmp( s1, s2, 7 ),
            "memcmp( s2, s1, 7 ) = ", memcmp( s2, s1, 7 ) );
    getch();
    return 0;
}
```

```
s1 = ABCDEFG
s2 = ABCDXYZ

memcmp( s1, s2, 4 ) = 0
memcmp( s1, s2, 7 ) = -1
memcmp( s2, s1, 7 ) = 1
```

String Kütüphanesindeki Diğer Fonksiyonlar

Prototip	Fonksiyon Tanımı
char *strerror(int errornum);	Sisteme bağımlı olmak üzere, hata sayısını bir metin stringi haline dönüştürür. Stringi gösteren bir gösterici döndürülür.
size_t strlen(const char *s);	s stringinin uzunluğunu bulur. null karakterden önceki karakterlerin sayısı döndürülür.

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    printf( "%s\n", strerror( 2 ) );
    getch();
    return 0;
}
```

```
No such file or directory
```

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main()
{
    const char *string1 = "abcdefghijklmnopqrstuvwxyz";
    const char *string2 = "dort";
    const char *string3 = "Boston";
    printf("\n%s \"%s %lu\n\"%s \"%s %lu\n\"%s \"%s %lu\n\"",
        string1, " uzunlugu", strlen( string1 ),
        string2, " uzunlugu", strlen( string2 ),
        string3, " uzunlugu", strlen( string3 ) );
    return 0;
}
```

```
"abcdefghijklmnopqrstuvwxyz" uzunlugu 26
"dort" uzunlugu 4
"Boston" uzunlugu 6
```


ÖRNEKLER

Soru 1: Klavyeden girilen bir cümlenin kelimelerini alt alta yazan program.

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

```
#include <string.h>
```

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

```
int 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");
```

```
        return 0;
```

```
    }
```

Soru 2

String1=Hello

String2=World

- ▶ String 1'i String 3'e kopyalayıp ekranda görüntüleyen
- ▶ String 1 ile String 2'yi birleştirip ekranda görüntüleyen
- ▶ String 1'in String 2 ile birleştirildikten sonraki uzunluğunu bulup ekranda yazdıran

C programı yazınız.

```
#include <stdio.h>
#include <string.h>
#include <conio.h>
int main ()
{
    char str1[12] = "Hello";
    char str2[12] = "World";
    char str3[12];
    int len ;
```

str1'in str3'e kopyalanması



```
strcpy(str3, str1);
printf("Cevap1: %s\n", str3 );
```

```
strcat( str1, str2);
printf("Cevap2: %s\n", str1 );
```

str1 ile str2'nin birleştirilmesi



```
len = strlen(str1);
printf("Cevap3: %d\n", len );
getch();
return 0;
}
```

str1'in uzunluğunun bulunması



Soru 3

- ▶ Klavyeden girilen şifrenin (**Şifre: math**)
 - ▶ doğru olup-olmadığını en fazla 3 denemede kontrol eden,
 - ▶ şifre doğru ise **‘Şifre Doğru’** mesajını ekranda görüntüleyen,
 - ▶ 3 denemede de başarısız olduğunda **‘Hakkınız Doldu’** mesajını ekranda yazdıran
- C programı yazınız.

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

```
int main()
{
    char sifre[20];
    int flag=0;
```

şifre doğru ise

```
for(i=1;i<=3;i++)
{
    printf ("sifre giriniz..");
    gets(sifre);
    if (strcmp(sifre,"math")==0)
    {
        flag=1;
        printf ("Sifre Dogru");
        break;
    }
    else
        printf ("\nSifre Yanlis");
}
if (flag==0)
    printf("Hakkiniz doldu");
}
```

Soru 4

- Klavyeden bir metin ve karakter girilerek, girilen karakterin metindeki sayısını bulan C programı yazınız.

```
#include <stdio.h>
#include <conio.h>
int main()
{
    char c[1000],ch;
    int i,k=0;
    printf("Ifade giriniz: ");
    gets(c);
    printf("Karakter giriniz: ");
    scanf("%c",&ch);
```

```
    for(i=0;c[i]!='\0';++i)
    {
        if(ch==c[i])
            ++k;
    }
    printf("%c karakterinden= %d tane", ch,k);
    getch();
    return 0;
}
```

Soru 5

- Klavyeden girilen metindeki, *harf sayısını*, *rakam sayısını*, *boşluk sayısını* bulup ekranda yazdıran C programı yazınız.

```
#include<stdio.h>
#include<conio.h>
int main()
{
    char w[150];
    int a=0,b=0,c=0,i;
    printf("Ifade giriniz:\n");
    gets(w);
```



```
for(i=0;w[i]!='\0';i++)
{
    if((w[i]>='a' && w[i]<='z') || (w[i]>='A'&& w[i]<='Z'))
        a++;
    else if(w[i]>='0' && w[i]<='9')
        b++;
    else if (w[i]==' ')
        c++;
}
printf("\nHarf sayisi: %d",a);
printf("\nRakam sayisi: %d",b);
printf("\nBosluk sayisi: %d",c);
getch();
return 0;
}
```

Soru 6


- Klavyeden girilen 10 kelimeyi alfabetik sırada sıralatan C programı yazınız.

```
#include<stdio.h>
#include<conio.h>
#include <string.h>
int main()
{
    int i,j;
    char str[10][50],temp[50];

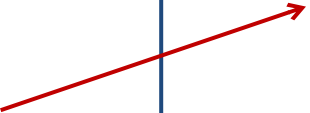
    printf("Enter 10 words:\n");
    for(i=0;i<10;++i)
        gets(str[i]);
```

```
for(i=0;i<9;++i)
    for(j=i+1;j<10 ;j++)
    {
        if(strcmp(str[i],str[j])>0)
        {
            strcpy(temp,str[i]);
            strcpy(str[i],str[j]);
            strcpy(str[j],temp);
        }
    }
printf("\nIn lexicographical order: \n");

for(i=0;i<10;++i)
{
    puts(str[i]);
}
getch();
return 0;
}
```



Kelimelerin alfabetik
sırada sıralanması



Sıralandıktan sonra
kelimelerin yazdırılması

SON