

ALGORİTMA VE PROGRAMLAMA I

Yrd. Doç. Dr. Deniz KILINÇ
deniz.kilinc@cbu.edu.tr

Genel Bakış...

2

- Karakter Dizileri
 - Karakter Dizilerini Okumak ve Yazmak
 - Karakter Dizilerinin Uzunluğunu Bulmak
 - Karakter Dizilerini Birleştirmek
 - Karakter Dizisi Kopyalamak
 - Karakter Dizilerini Karşılaştırmak
- Çok Boyutlu Diziler

7. BÖLÜM

3

Karakter Dizileri

Karakter Dizileri (Strings)

4

- Bazı programlama dillerinde karakter dizilerini tutmak için özel veri türleri (string, vb.) bulunmaktadır.
- Ancak C programlama dilinde böyle bir veri türü olmadığı için yerine **karakterlerden oluşan bir boyutlu diziler** kullanılır.
- Karakter dizilerine özel olarak, karakter dizilerinin sonuna **sonlandırıcı karakter** olarak adlandırılan bir simge eklenir.
- Sonlandırıcı karakter:
 - Dizinin bittiği yeri gösterir.
 - ASCII tablosunun sıfır numaralı ('\0') karakteridir.

Karakter Dizileri (Strings) devam...

5

- Karakter dizilerine 2 şekilde başlangıç değeri verilebilir:
 - (1) `char s[7] = {'d', 'e', 'n', 'e', 'm', 'e', '\0'};`
 - (2) `char s[7] = "deneme";`
- Birinci tanımlamada sonlandırıcı karakter programcı tarafından konmalıdır.
- İkinci tanımlamada ise buna gerek yoktur. Çünkü, sonlandırıcı karakter bu atamayla, derleyici tarafından eklenir.

Karakter Dizilerini Okumak

6

- Bir karakter dizisini klavyeden okumak için C'nin standart **gets()** fonksiyonu kullanılır.
- **stdio.h** dışında yeni bir kitaplığı C programına dahil etmeye gerek yoktur.
- Bu fonksiyon herhangi bir indeks tanımlamadan karakter dizilerinin okunmasını sağlar.
- Okuduğu karakter dizisinin sonuna satır sonu işaretini değil, **NULL** değerini yerleştirir.

Karakter Dizilerini Okumak (devam...)

7

- `gets()` fonksiyonu, klavyeden girilen karakter dizilerini, herhangi bir ek tanımlamaya gerek duymadan bir dizi içine yerleştirir.
- Dizinin her bir karakteri dizinin farklı bir hücresi içine yerleşir.
- **Örnek:**

```
char ad[20];
```

```
...
```

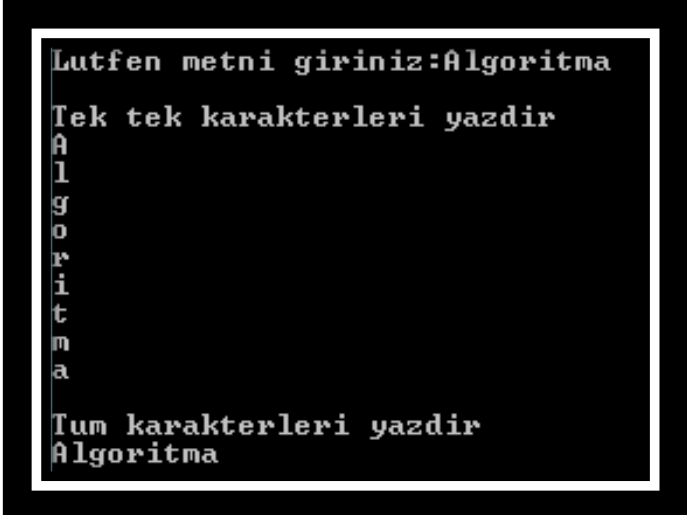
```
gets(ad);
```

Örnek-5:

Karakter Dizisi Okuma ve Yazma

8

- Maksimum 50 karakter okuyabilecek bir karakter dizisi tanımlayın.
- Klavyeden karakter dizisini okuyun ve ekrana karakterleri yazdırın.
 - Tek tek karakterleri yazdırın
 - Tüm metni tek seferde yazdırın



```
Lutfen metni giriniz:Algoritma
Tek tek karakterleri yazdir
A
l
g
o
r
i
t
m
a

Tum karakterleri yazdir
Algoritma
```


Örnek-5:

Karakter Dizisi Okuma ve Yazma (devam...)

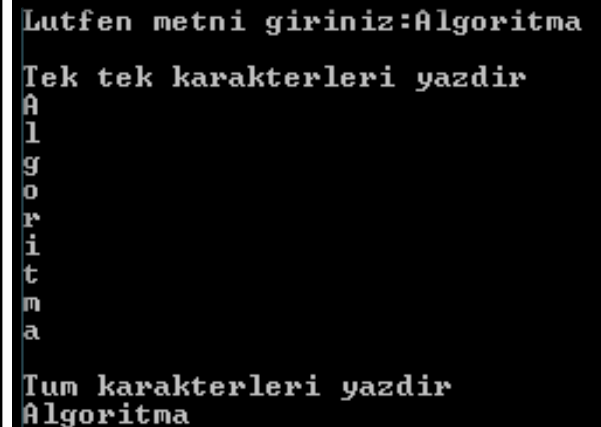
9

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

int main()
{
    char dizi[50];
    int i;
    printf("Lutfen metni giriniz:");
    gets(dizi);

    printf("\nTek tek karakterleri yazdir\n");
    for (i=0; dizi[i]; i++){
        printf("%c \n", dizi[i]);
    }

    printf("\nTum karakterleri yazdir\n");
    printf(dizi);
    printf("\n");
    return 0;
}
```



```
Lutfen metni giriniz:Algoritma
Tek tek karakterleri yazdir
A
l
g
o
r
i
t
m
a

Tum karakterleri yazdir
Algoritma
```

Karakter Dizilerinin Uzunluğu Bulmak

10

- Bazı uygulamalarda bir karakter dizisinin uzunluğunu bulmak gerekebilir.
- Bir karakter dizisinin uzunluğunu, yani kaç karakter içerdiğini bulmak için C'nin standart `strlen()` fonksiyonu kullanılır.
- Uzunluk bulunurken, içerdiği en son karakter olan **NULL** karakteri göz önüne alınmaz.
- Örneğin, karakter dizisi `"abc"` değerlerini içeriyorsa, `strlen()` fonksiyonu bu uzunluk olarak `"3"` değerini döndürür.

Örnek-6:

Girilen Karakter Dizisinin Uzunluğunu Bulmak

11

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

int main()
{
    char dizi[50];
    int i;
    printf("Lutfen metni giriniz:");
    gets(dizi);

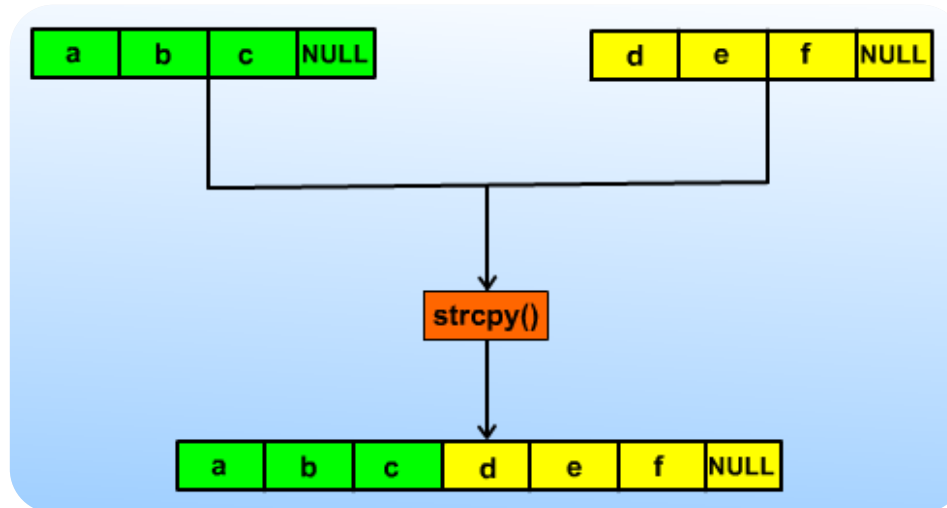
    printf("Uzunluk:%d", strlen(dizi));
    printf("\n");
    return 0;
}
```

```
Lutfen metni giriniz:Algoritma
Uzunluk:9
```

Karakter Dizilerini Birleştirmek

12

- İki karakter dizisini birleştirilerek tek bir karakter dizisi haline dönüştürmek için C'nin `strcat()` fonksiyonu kullanılır.
- Bu fonksiyon, var olan bir karakter dizisinin sonuna bir başka karakter dizisini ekleyecektir.
- Örneğin `"abc"` karakter dizisinin sonuna `"def"` karakter dizisi `strcat()` fonksiyonu kullanılarak eklenebilir.



Örnek-7:

Girilen Karakter Dizilerini Birleştirmek

13

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

```
int main()
```

```
{
```

```
    char metin1[50], metin2[50];
```

```
    printf("Lutfen 1.metni giriniz:");
    gets(metin1);
```

```
    printf("Lutfen 2.metni giriniz:");
    gets(metin2);
```

```
    strcat(metin1, metin2);
```

```
    printf(metin1);
    printf("\n");
    return 0;
```

```
}
```

```
Lutfen 1.metni giriniz:Algoritma
Lutfen 2.metni giriniz:ve Programlama
Algoritmave Programlama
```

Karakter Dizisi Kopyalama

14

- Karakter dizilerine direk atama yapılamamaktadır.
- Örneğin aşağıdaki atama ifadesi yanlıştır:

```
char dizi[50];  
katar = "abcde";
```

- Çünkü bu atama göstergeye yapılan atamadır.
- Göstergelerin ne olduğunu daha sonra detaylı olarak ele alınacaktır.
- Atamanın bir karakter dizisine yapılabilmesi için, C'nin standart `strcpy()` fonksiyonu kullanılır.

Örnek-8:

Karakter Dizilerini Kopyalamak

15

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

int main()
{
    char metin1[50], metin2[50];

    strcpy(metin1, "Algoritma ");
    strcpy(metin2, "ve Programlama ");

    strcat(metin1, metin2);

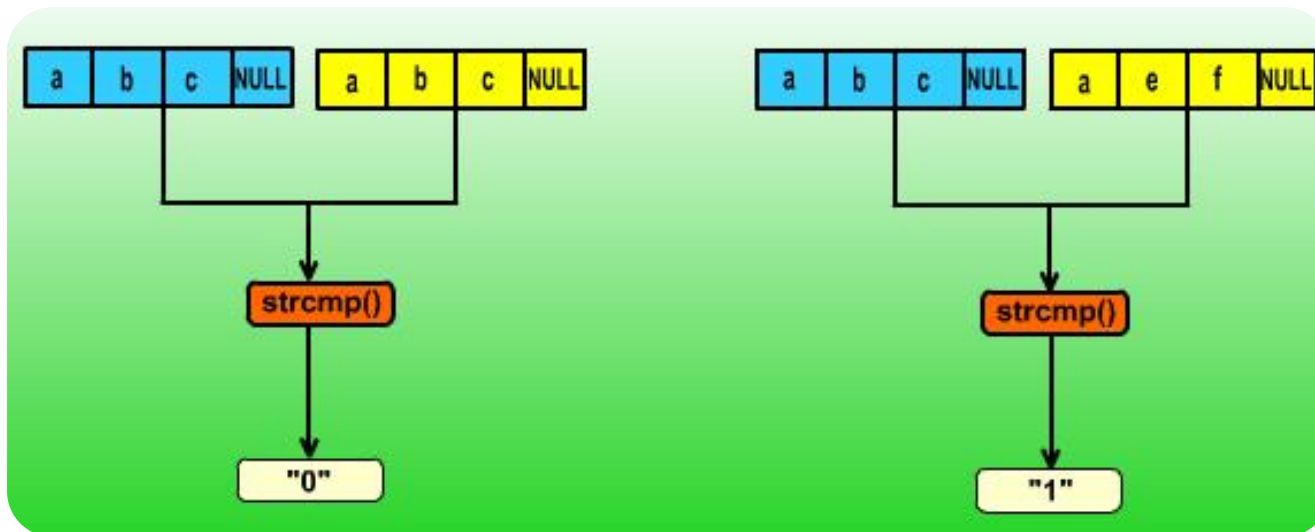
    printf(metin1);
    printf("\n");
    return 0;
}
```

Algoritma ve Programlama

Karakter Dizilerini Karşılaştırmak

16

- İki karakter dizisinin birbirleriyle karşılaştırılarak, içerdiği karakterlerin aynı olup olmadıkları test edilebilir.
- Bu amaçla `strcmp()` fonksiyonu kullanılır.
- Karşılaştırma sonucunda, **her iki karakter dizisi birbirinin aynı ise "0"**; **birbirinden farklı ise "1"** değeri üretilir. Elde edilen bu değer kullanılarak programın akışı yönlendirilebilir.



Örnek-9:

Karakter Dizilerini Karşılaştırmak

17

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

int main()
{
    char metin1[50], metin2[50];

    strcpy(metin1, "Algoritma ");
    strcpy(metin2, "ve Programlama ");

    if (strcmp(metin1, metin2))
        printf("Metinler birbirinden farklı...");
    else
        printf("Metinler birbirleriyle aynı...");

    printf("\n");
    return 0;
}
```

Metinler birbirinden farklı...

Örnek-10:

Girilen Cümleyi Tersten Yazdırma

18

- 100 elemanlı bir karakter dizisi tanımlayınız.
- Daha sonra bir cümle giriniz.
- Cümleyi ekrana ters olarak yazdırınız.
- İsterseniz farklı bir diziye aktarabilirsiniz veya direk yazdırabilirsiniz.
- **strrev()** fonksiyonunu kullanmayınız.

Örnek-10:

Girilen Cümleyi Tersten Yazdırma

19

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

int main()
{
    char cumle[100];
    int i;

    printf("Lutfen cumle giriniz: ");
    gets(cumle);

    for (i=strlen(cumle); i>=0; i--)
    {
        printf("%c", cumle[i]);
    }
    return 0;
}
```

Örnek-11:

Girilen Cümlede Harf Arama ve Yerini Bulma

20

- 100 elemanlı bir karakter dizisi tanımlayınız. Daha sonra bir cümle giriniz.
- Cumlede aranmak üzere bir harf giriniz.
- Cümlede harfi bulursanız, bulduğunuz yerlerin pozisyonlarını farklı bir diziye aktarınız.
- Toplam bulunan harf sayısını ve harflerin pozisyonlarını ekrana yazdırınız.

```
Lutfen cumleyi giriniz:deniz emin sibel  
Lutfen aranacak harfi giriniz:e  
Toplam bulunan harf sayisi: 3  
Cumledeki pozisyon: 1  
Cumledeki pozisyon: 6  
Cumledeki pozisyon: 14
```

Örnek-11:

Girilen Cümlede Harf Arama ve Yerini Bulma

21

```
int main()
{
    char cumle[100];
    char harf;
    int i, harfSayisi = 0, harfPozisyon[50];
    if (harfSayisi == 0)
    {
        printf("Aradiginiz harf bulunamadi...");
    }
    else
    {
        printf("Toplam bulunan harf sayisi: %d\n", harfSayisi);
        for (i = 0; i < harfSayisi; i++)
            printf("Cumledeki pozisyon: %d \n", harfPozisyon[i]);
    }
    if
    {
        return 0;
    }
    {
        harfPozisyon[harfSayisi] = i;
        harfSayisi++;
    }
}
```

Örnek-12:

Girilen Bir Cümlenin Kelimelerinin Baş Harflerini Büyük Harf Yapma

22

- 100 elemanlı bir karakter dizisi tanımlayınız. Daha sonra bir cümle giriniz.
- Cumledeki tüm kelimelerin baş harflerini büyük harfe çeviriniz. Büyük harfe çevirme için **toupper()** fonksiyonu kullanılacaktır.
- Çeviride Türkçe karakterler desteklenmeyecektir.
- Son olarak cümlenin çevrilmiş halini ekrana yazdırınız.

```
Lutfen cumleyi giriniz:deniz sibel ada  
Cumle:  Deniz Sibel Ada
```

Örnek-12:

Girilen Bir Cümlenin Kelimelerinin Baş Harflerini Büyük Harf Yapma

23

```
int main()
{
    char cumle[100];
    int i;

    printf("Lutfen cumleyi giriniz:");
    gets(cumle);

    for (i = 0; i < strlen(cumle); i++)
    {
        if (cumle[i] == ' ')
            cumle[i+1] = toupper(cumle[i+1]);
        else
            if (i == 0)
                cumle[i] = toupper(cumle[i]);
    }

    printf("\nCumle:  %s", cumle);
    return 0;
}
```

KAYNAKLAR

24

- N. Ercil Çağıltay ve ark., C DERSİ PROGRAMLAMAYA GİRİŞ, Ada Matbaacılık, ANKARA; 2009.
- Milli Eğitim Bakanlığı "Programlamaya Giriş ve Algoritmalar Ders Notları", 2007
- <http://tr.wikipedia.org/wiki/Code::Blocks>
- <http://www.codeblocks.org>
- <http://www.AlgoritmaveProgramlama.com>
- <http://www1.gantep.edu.tr/~bingul/c>



Algoritma ve Programlama

İYİ ÇALIŞMALAR...

Yrd. Doç. Dr. Deniz KILINÇ
deniz.kilinc@cbu.edu.tr