BLM1011 BİLGİSAYAR BİLİMLERİNE GİRİŞ

GR.2

2023-2024 GÜZ YARIYILI

DR.ÖĞR.ÜYESİ GÖKSEL BİRİCİK

Karakter Dizileri

char Sabitleri

Karakterin tek tırnak (single quote) içinde yazılması ile:

‘a’ ‘X’ ‘>’ …

Tek tırnak içindeki karakter sabiti aslında o karakterin ASCII tablosundaki sıra numarasını gösteren sayıdır.

```
char ch;

ch = ‘a’;          //ch aslında 97.   (0110 0010)

ch = ‘a’ + 1;      // ch aslında 98. karakter olarak, b (0110 0011)

ch = ‘1’;          //ch aslında 49.   (0011 0001)
```

Öntanımlı char Sabitleri

Ekrana basılamayan karakterleri sabit hale getirmek için \ kullanılır.

'\a'	ASCII 7	Bip
'\b'	ASCII 8	Geri boşluk (backspace)
'\f'	ASCII 12	Sayfa ileri (form feed)
'\n'	ASCII 10	Aşağı satır (new line)
'\r'	ASCII 13	Satır başı (carriage return)
'\t'	ASCII 9	Tab
'\v'	ASCII 11	Düşey tab (vertical tab)
'\\'	\	
'\''	'	
'\0'	ASCII 0	NULL karakter

Diğer Tabanlarda char Sabitleri

‘\x41’ ASCII 41H

‘\012’ ASCII 10

‘\0123’ ASCII 66

```
char a,b;
a = ‘\xff’;
b = ‘\0123’;
…
```

ASCII 7:

‘\x7’

‘\07’

‘\a’ // tercih sebebidir.

Karakter Dizisine İlk Değer Verme

Karakter dizisi:

‘a’	‘b’	‘c’	‘d’	‘e’

⇔ ASCII Tablosu

97	98	99	100	101

char y[10] = { ‘Y’, ’I’, ’L’, ’D’, ’I’, ’Z’, ’\0’ };

- NULL karakteri biz sona eklemeliyiz.

char t[] = { ‘T’, ’E’, ’K’, ’N’, ’İ’, ’K’, ’\0’ };

- t[0] = ‘T’ t[1] = ‘E’ …

char u[] = "ÜNİVERSİTESİ";

- NULL Karakteri sona derleyici kendisi yerleştirir.

Sonlandırıcı Karakter Ne İşe Yarıyor?

Karakter dizisinin uzunluğunun bilinmesi gerekliliğini ortadan kaldırır.

Karakter dizilerini kullanan algoritmalar dizinin sonlandırıcı karakter ile bittiği varsayarak çalışırlar.

Karakter dizisini fonksiyona parametre olarak aktarırken uzunluğunu da bildirmek zorunda kalmayız.

n elemanlı bir diziye en fazla n-1 karakter yerleştirebiliriz. (+ ‘\0’)

Karakter Dizisi Okuma ve Yazma

```
char s[20];
scanf("%s",s);      // ‘\0’ karakterini kendisi koyar. Dizi zaten işaretçi adresi olduğu için & yok.
gets(s);            // ‘\0’ karakterini kendisi koyar. //C11’de kaldırıldı
puts(s);            //printf("%s\n",s); ile eşdeğerdir.
```

gets() ile okuduğunuz değer, dizinin boyutundan fazla ise, taşma durumu oluşur.

Derleme zamanı değil, çalışma zamanı hatasıdır. Programınız hata ile sonlanır.

puts() (ya da %s ile printf) sonlandırıcı karakteri görene kadar tüm karakterleri ekrana yazar. Herhangi biçimde sonlandırıcı karakter ezilirse bu fonksiyonlar tesadüfen ilk sonlandırıcı karakteri görene kadar ekrana yazmaya devam ederler.

Yazının Uzunluğunu Bulma

```
#include <stdio.h>
int main()
{
        int k = 0;
        char s[50];

        scanf("%s",s);
        while(s[k]!='\0')
           k++;

        printf("uzunluk = %d \n",k);
        return 0;
}
```

Yazıyı Ters Çevirme

```
#include <stdio.h>
int main()
{
		int n, j, temp;
		char s[50];

		 scanf("%s",s);
		for(n=0;s[n];n++);

		for(j=0;j<n/2;j++){
		   temp = s[n-j-1];
		   s[n-j-1] = s[j];
		   s[j] = temp;
		}

		 printf("%s",s);
		return 0;
}
```