



BİLGİSAYAR PROGRAMLAMA 1

Ders Notu 3– C Fonksiyonlarına Giriş ve Değişkenler

Konya Teknik Üniversitesi
Elektrik – Elektronik Mühendisliği Bölümü

15.03.2024

Konya

C Dilinin Özellikleri

- C dili donanımdan bağımsız (machine independent) ve taşınabilirdir (portable).
- Taşınabilirlik, aynı programın farklı donanımlarda ve farklı işletim sistemlerinde işletilebilirliğini ifade eder.
- Bir dilin verimliliği, hızlı fakat fazla yer kaplamayan yazılımlar geliştirmeye olanak sağlaması ile ölçülür.
- C Programlama dili verimli, basit, fakat güçlü yapısından ve bu dilde geliştirilen uygulamaların taşınabilir olmasından dolayı yaygın olarak kullanılmaktadır.

C Programlarının Temel Yapısı

Bir C programı iki temel bölümden oluşur;

- Önışlemci Direktifleri (Preprocessor Directives)
- Ana Fonksiyon (Main Function)

Önişlemci Direktifleri

- Önişlemci direktifleri # işareti ile başlar ve program derlenmeden önce C önişlemcisi tarafından işletilir.
- Her bir önişlemci direktifinin farklı bir görevi vardır.
- **#include** ve **#define** en çok kullanılan direktiflerdir.
- **#include** direktifi program içerisinde kullanılan fonksiyonlar için gerekli kodları programa dahil etmek için kullanılır.

Önişlemci Direktifleri

- Tasarladığımız programda ekrana çıktı yazdırmak istiyorsak; C dilinin standart bir fonksiyonu olan **printf** fonksiyonunu kullanmamız gerekir.
 - **printf(“Örnek Çıktı”);**
- Ancak **printf** fonksiyonunun çalışabilmesi için **<stdio.h>** isimli başlık dosyasına ihtiyacımız olacaktır.
- Bu dosyayı programa dahil etmek için program kodunun en tepesine **#include<stdio.h>** komut satırı yazılır.
- C dilinde **.h** uzantılı dosyalara başlık dosyası (**HeaderFile**) adı verilir.
- **stdio.h** başlık dosyası standart giriş çıkış (**Standard input-output**) işlemleri için gerekli kodları içerir.

Bazı Kütüphane Başlık Dosyaları

<u><ctype.h></u>	Karakter Fonksiyonları
<locale.h>	Konumlandırma Fonksiyonları
<u><math.h></u>	Matematik Fonksiyonları
<setjmp.h>	Atlama Fonksiyonları
<signal.h>	Sinyal Yönetme Fonksiyonları
<stdarg.h>	Değişken argümanları Yönetme Fonksiyonları
<stdio.h>	Standart Giriş/Çıkış Fonksiyonları
<stdlib.h>	Standart Fayda Fonksiyonları
<u><string.h></u>	String Yönetme Fonksiyonları
<time.h>	Zaman Fonksiyonları

Bazı Kütüphane Başlık Dosyaları

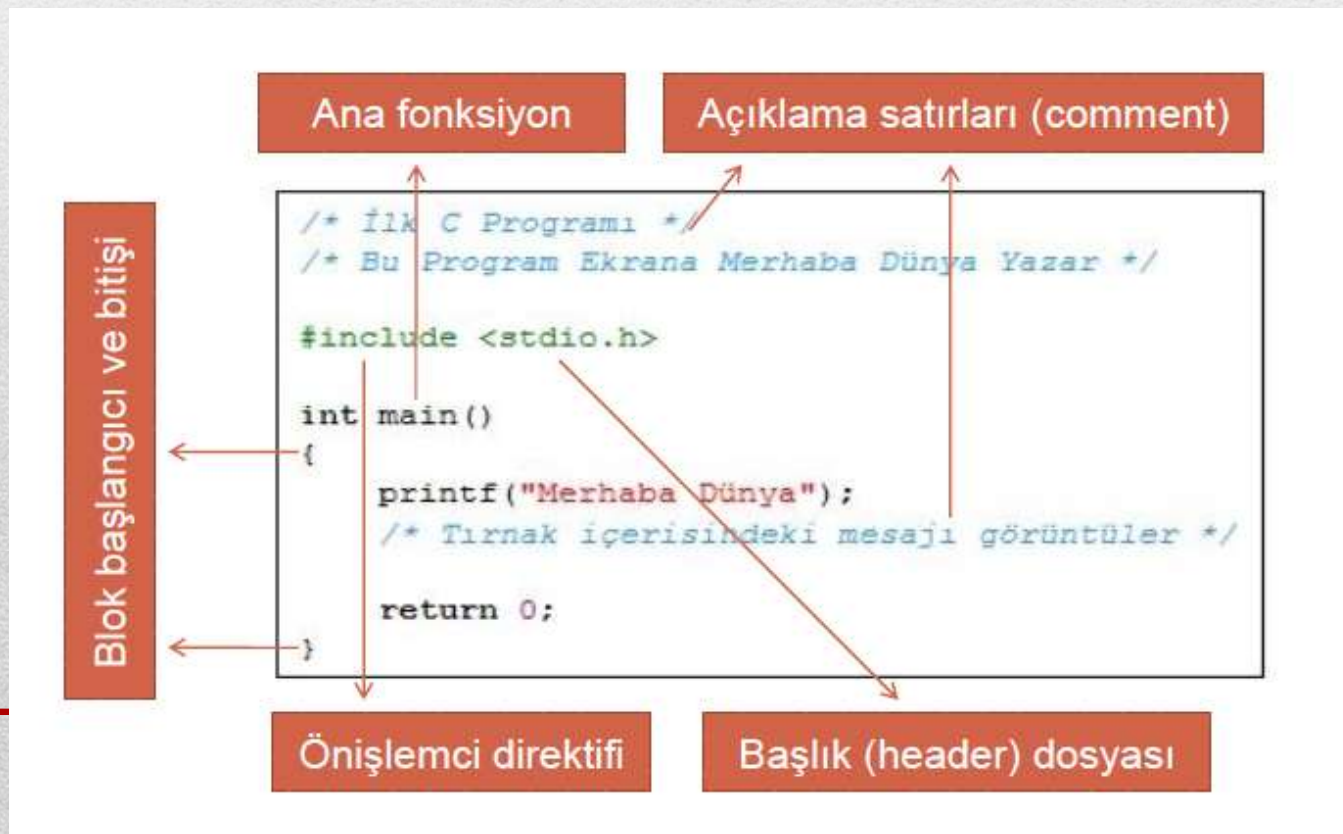
- `<stdlib.h>`
 - `exit`: Çalışan programı durdurur
 - `qsort`: Diziyi sıralar
 - `div`: Tamsayıyı böler
 - `<stdio.h>`
 - `printf()`: character, string, float, integer, octal ve hexadecimal türündeki değişkenlerin değerlerini ekrana yazdırır
 - `getch()`: Dosyadan karakter okur
 - `fopen()`: Okuma, yazma gibi değişik işlemleri gerçekleştirmek için bir dosyayı açar
 - `<math.h>`
 - `sin()`: sinüs değerini hesaplar
 - `asinh()`: hiperbolik arc sinüs değerini hesaplar
 - `sqrt()`: bir sayının karekök değerini hesaplar
-

Genel giriş-çıkış kütüphanesi

- Geliştirilen bir programda kullanıcıdan bilgi (veri) almak ve kullanıcıya bilgi göstermek en temel kullanıcı etkileşim işlemleridir.
- C dilinde bütün giriş-çıkış operasyonları giriş ve çıkış fonksiyonları (input-output functions) ile gerçekleştirilir.
- C de kullanılan standart giriş-çıkış fonksiyonları, **printf()** ve **scanf()** dir.
- Bu fonksiyonlar C standart giriş-çıkış başlık dosyasında **stdio.h** bulunurlar.
- Giriş-çıkış fonksiyonlarını program içerisinde kullanabilmek için **stdio.h** başlık dosyasının programa dahil edilmesi gerekir.
- `#include<stdio.h>`

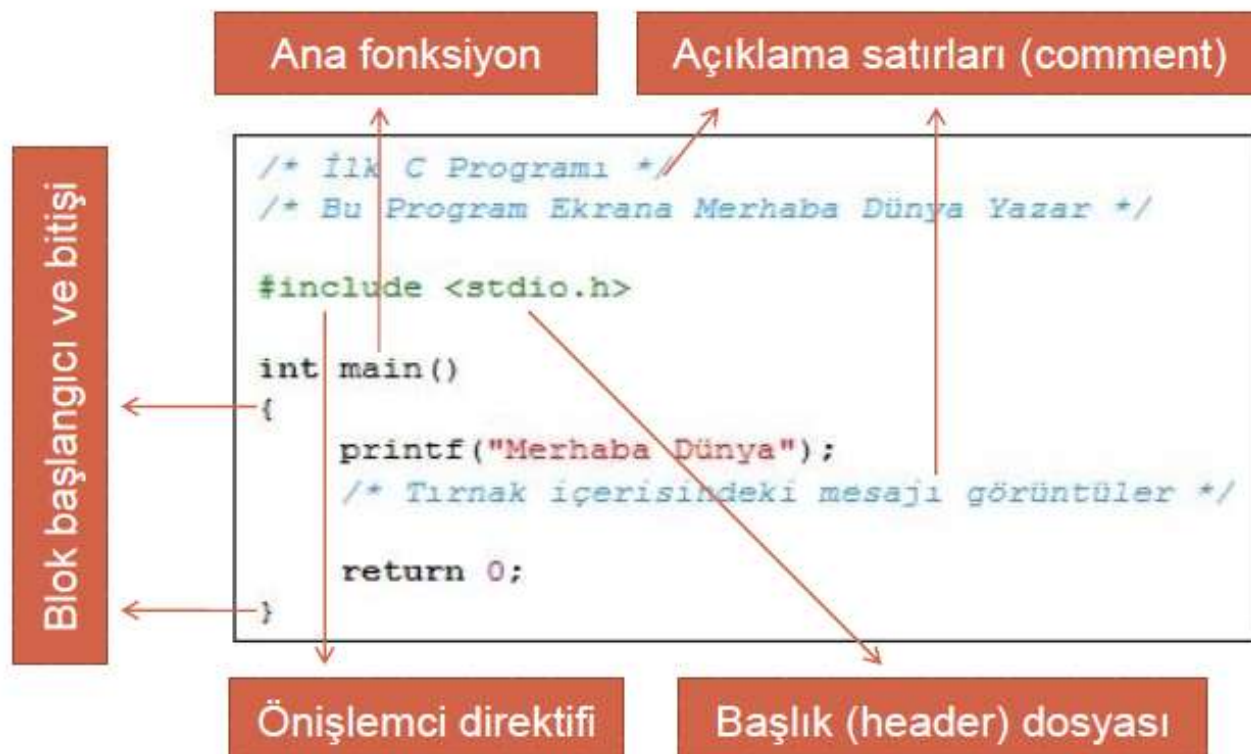
Main Fonksiyonu

- Hemen hemen bütün C programları birden fazla fonksiyondan oluşur.
- main() bütün C programlarında bulunması gereken programın ana fonksiyonu yani gövdesidir.
- Program koşturulduğunda ilk çalıştırılacak olan fonksiyondur.



Main Fonksiyonu

- Fonksiyonda çalıştırılacak ifadeler (kod satırları) { } küme parantezleri içinde yazılırlar.
- Her parantez çiftinin oluşturduğu yapılar kod blokları denir.
- Bir kod bloğu içerisinde program içerisinde kullanılacak değişkenler ve gerçekleştirilecek işlemleri yerine getirecek komutlar bulunur.



printf() fonksiyonu

- Değişkenlerin değerlerini, hesaplanan sonuçları ya da mesajları ekranda göstermek için kullanılır.
- printf() fonksiyonu, fonksiyon ismi ve parantezler içindeki parametreler olmak üzere iki kısımdan oluşur.
- printf() fonksiyonu, parametre olarak görüntülenecek bilginin hangi biçimde görüntüleneceğini bildiren çıktı metin formatını ve bu formatın içinde yazdırılacak olan değişkenler listesini alır.
- Genel kullanım formatı aşağıdaki gibidir.
 - printf(“Merhaba Dünya”);
 - printf(“çıkıttı metin formatı”, değişken listesi)
 - printf(“Hesaplanan Yaşınız: %d dir”, yas);

printf() fonksiyonu

- `printf(“Hesaplanan Yaşınız: %d dir”, yas);`
- `printf` çıktı formatındaki `%d` karakterlerinin dışındaki tüm karakterleri ekrana yazar.
- `%` karakterini gördüğünde bunun sağındaki karakteri yazdırılacak değişkenin format karakteri olarak ele alır.

Format Karakterleri

Format Karakterleri	Anlamı
d	Tamsayılar için (onluk sistemde)
i	Tamsayılar için (onluk, sekizlik ve onaltılık sistemde)
lx	Uzun tamsayıları onaltılık (hex) sistemde
ld	Uzun tamsayılar için (onluk sistemde)
o	Tamsayılar için (sekizlik sistemde)
x	Tamsayılar için (onaltılık (hex) sistemde)
X	Tamsayılar için (onaltılık (hex) sistemde büyük harfle)
u	İşaretsiz tamsayılar için
c	Char tipinde değişkenler (karakterler) için
s	Metin (string)
f	Float değişkenler için
E , e	Double ve float türlerini üstel gösterimde yazmak için
lf	Double değişkenler için

printf() fonksiyonu

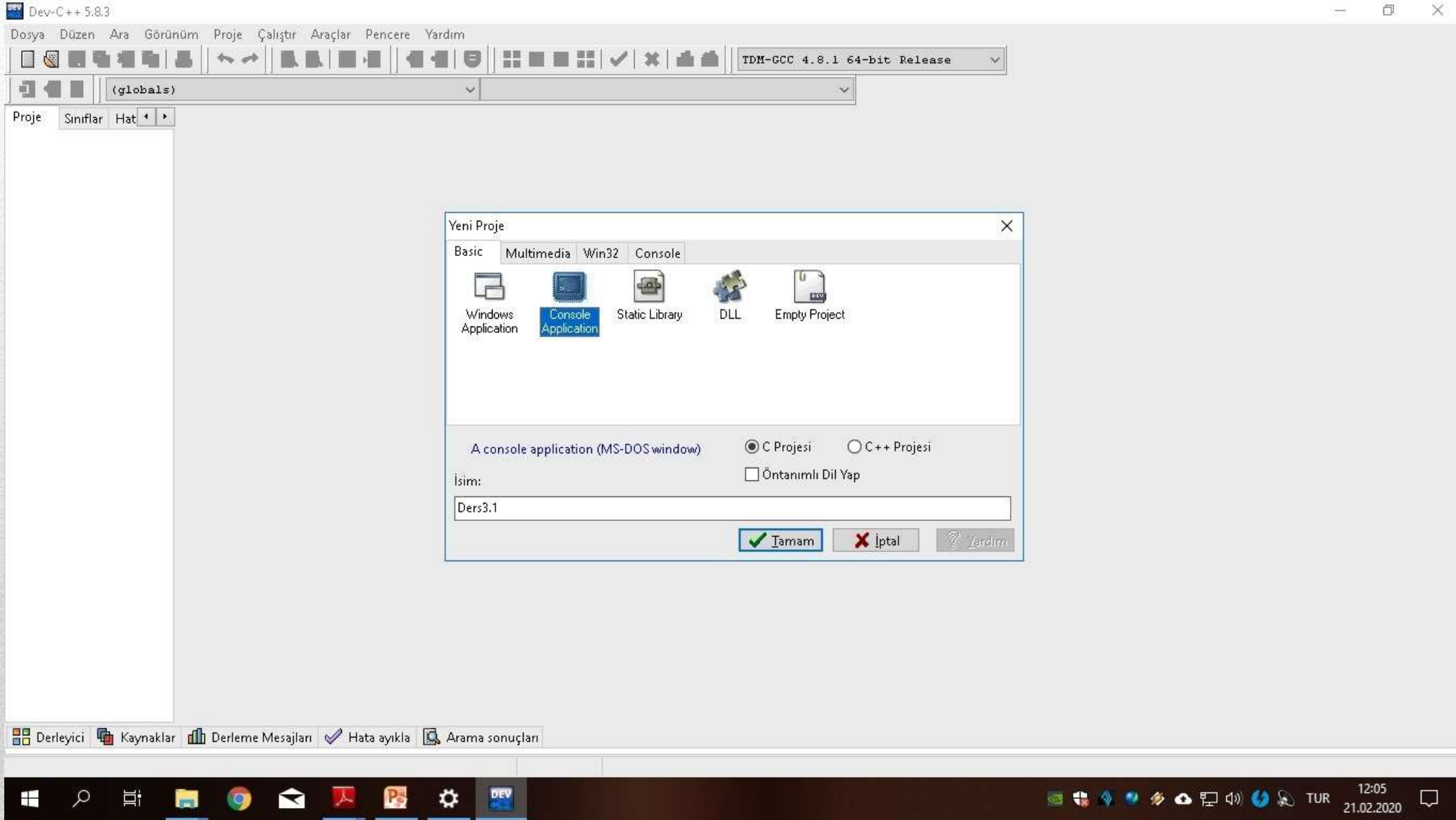
- printf(“Hesaplanan \nYaşınız: %d dir”, yas);
- printf çıktısı formatındaki \ karakterine escape karakteri denir.
- Bu karakterden sonra gelen karakter ise escape serisini ifade eder.
- Örneğin \n ifadesi, çıktı ekranında yeni bir satıra geçilmesi gerektiğini ifade eder.

\0	null karakteri temsil eder (sonlandırıcı karakter)
\n	Yeni satır
\r	Satırbaşı
\t	Yatay sekme
\v	Düşey sekme
\f	İleri besleme
\b	Geri boşluk (space)
\a	Alarm karakteri
\"	Çift tırnak
\\	Ters bölü

C dilinin genel yazım kuralları

- C dili birden fazla satırdan oluşan açıklama satırlarına izin verir.
- Bu açıklama satırları programın ne iş yaptığı hakkında bilgi verir.
- Açıklama satırları /* ile başlayıp */ile sona erer.
- Derleyici bu satırları çalışma anında dikkate almaz.
- C’de her bir işletilebilir ifade (komut satırı) ; ile sonlandırılır.
- Bütün anahtar kelimeler ve komutlar küçük harfle yazılır (**#define** hariç. Define ile yapılan tanımlamalarda istenirse büyük harf kullanılabilir).
- C dili büyük-küçük harf duyarlıdır. Yani; “TOPLAM”, “toplam” ve “tOpLaM” kelimelerinin hepsi C derleyicisi tarafından ayrı ayrı algılanır.

İlk C Programınız



İlk C Programınız

- `#include <stdio.h>`
- `#include <stdlib.h>`
- `/* run this program using the console pauser or add your own getch, system("pause") or input loop */` => **Yorum (Açıklama)**
- `int main()`
- `{`
- `printf("Merhaba Dunya");`
- `getch();`
- `return 0;`
- `}`

Değişken Tanımlamaları ve Veri Türleri

- Programda kullanılacak bütün değişkenlerin kullanılmadan önce C derleyicisine bildirilmesi gerekir.
- Değişkenleri derleyiciye bildirme işine değişken tanımlama (variable definition) denir.
- Değişken tanımlamak, değişken için bir isim seçmek ve içerisinde ne tür bir veri saklanacağını söylemekle yapılır.
- Her değişkenin ismi, veri türü ve değeri vardır.
- Değişken ismi değişken saklayacak bellek hücrelerine verilecek isimdir.
- Veri türü saklanan bilginin türüdür (ondalık sayı, tam sayı..).
- Değeri ise değişken içinde saklanan bilginin kendisidir.

Değişken İsimlendirme

- Programda kullanılacak değişken isimleri programcı tarafından tanımlanır.
- C dilinin kendine özgü anahtar sözcükleri, komut veya fonksiyon adları değişken ismi olarak kullanılamaz.
- C dilinin bazı Anahtar Sözcükleri (ISO/IEC 9899:1999):

auto	do	goto	return	union	_Imaginery
break	double	if	short	unsigned	
case	else	inline	signed	void	
char	enum	int	sizeof	volatile	
const	extern	long	static	while	
continue	float	register	struct	_Bool	
default	for	restrict	typedef	_Complex	

Değişken İsimlendirme

- Değişken isimleri içerisinde, a-z ve A-Z arası İngilizce harfleri, 0-9 arası rakamlar ve özel karakter olarak sadece alt çizgi (_) karakteri kullanılabilir.
- Özel karakterler (+,-,! vs.) ve Türkçe karakterler kullanılmaz.
Örn: Maaş, öğrenci, sınıf => bunlar değişken adı olarak kullanılamaz.
- Değişken ismi rakam ile başlayamaz.
Örn: 1.ogrenci => (yanlış) ogrenci1 => (doğru)

Veri türleri

- Veri türü, değişken içerisinde tutulan değerin tipini belirler (tamsayı, ondalık sayı, karakter, vs.).
- C dilinde dört temel veri tipi (türü) vardır.
- **char, int, float ve double.**
- **char:** bir harf, bir rakam veya özel bir simgeyi saklamak için kullanılan veri tipidir.
- **char** tipindeki her bir değer tek tırnak arasına yazılmalıdır
- ("A", "z", "2", "*", ":", " ").
- **char** tipinde tanımlanan bir değişkenin tuttuğu değer, karakterin ASCII tablosundaki sayısal karşılığıdır.

Veri türleri

- **int:** tam sayı değerlerini saklamak için kullanılır.
- **float:** ondalıklı olarak verilen sayıları ifade edebilmek için kullanılır.
- **double:** ondalıklı olarak verilen çok küçük ve çok büyük sayıları ifade edebilmek için kullanılır. Float'a göre iki kat daha fazla duyarlılığa sahiptir.

Veri türleri

Tip	Sınırlar			Bellekte alanı
char	-128	...	+127	8 bit
unsigned char	0	...	+255	8 bit
signed char	-128	...	+127	8 bit
int	-32768	...	+32767 veya	16 bit veya
	-2147483648	...	+2147483647	32 bit
unsigned int	0	...	+65535 veya	16 bit veya
	0	...	+4294967294	32 bit
signed int	-32768	...	+32767 veya	16 bit veya
	-2147483648	...	+2147483647	32 bit
short int	-32768	...	+32767	16 bit
unsigned short int	0	...	+65535	16 bit
signed short int	-32768	...	+32767	16 bit
long int	-2147483648	...	+2147483647	32 bit
signed long int	-2147483648	...	+2147483647	32 bit
unsigned long int	0	...	+4294967294	32 bit
float	$\pm 3.4\text{E}-38$...	$\pm 3.4\text{E}+38$	32 bit
double	$\pm 1.7\text{E}-308$...	$\pm 1.7\text{E}+308$	64 bit
long double	$\pm 1.7\text{E}-308$...	$\pm 1.7\text{E}+308$	64 bit veya
	$\pm 3.4\text{E}-4932$...	$\pm 1.1\text{E}+4932$	80 bit

Değişken tanımlama yerleri, şekilleri ve ilk değer atama

- Standart C'de üç farklı yerde değişken tanımlanabilir.
- fonksiyonların üstünde, blokların { } içerisinde ilk sırada ve fonksiyonlarda parametre olarak.
- İlk değer ataması yapılmayan değişkenlerin değerleri (eğer main fonksiyonunun üstünde tanımlanmışsa) sayısal olanlar 0 diğerleri boş olarak belirlenir, eğer main içinde tanımlanmışsa bellekte rastgele değerler olarak belirlenir.

```
#include<stdio.h>

/* Örnek Değişken tanımlamaları */

int main() {

    int a;
    a=1;

    int x, y, z; /* Aynı satırda birden fazla değişken tanımlanabilir. */

    char m='k';

    double n=5.05;

}
```



ders3.2.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <math.h>
4  /* run this program using the c
5  int a;
6  float b;
7  double c;
8
9  int main()
10 {
11     a=M_PI;
12     printf("\n %d",a);
13
14     b=M_PI;
15     printf("\n %f",b);
16
17     c=M_PI;
18     printf("\n %lf",c);
19
20     float x=M_PI;
21     printf("\n %.12f",x);
22
23     double y=M_PI;
24     printf("\n %.12lf",y);
25
26     float z=4.79;
27     printf("\n %f",z);
28
29     double m=4.79;
30     printf("\n %lf",m);
31
32     return 0;
33 }
```

- ÖRNEK: Pi Sayısı=3.141592 653589 793...
- float veri türü, virgülden sonra en fazla 6 basamak için kullanılabilir.

C:\Users\Kemal\Desktop\C_Kemal_Ders\Kodlar\ders3.2.exe

```
3
3.141593
3.141593
3.141592741013
3.141592653590
4.790000
4.790000
```

Process exited after 0.01311 seconds with return value 0
Press any key to continue . . .

scanf() fonksiyonu

- Değişkenlerin içerisine klavyeden değer atamak için kullanılır.
- Fonksiyon ismi ve parametrelerden oluşur.
- Parametre olarak, girilecek değer hangi formatta olacağını bildiren girdi formatını ve bu formata göre girilecek değişkenler listesini alır.
- scanf fonksiyonunda dışarıdan değer girilecek bütün değişkenlerin başına & işareti konur.
- Bu işaret bellek operatörüdür, değişkenlerin tutulduğu bellek hücresinin adresini işaret eder.
- ÖRN:

```
int t;  
scanf("%d",&t);
```


Örnek

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* run this program using the console pauser or add your own getch, system("pause") or input loop */
5  int y;
6  int t,a;
7
8  int main()
9  {
10     printf("\nUcgenin yukseklik degerini giriniz");
11     scanf("%d",&y);
12     printf("\nUcgenin taban uzunlugu degerini giriniz");
13     scanf("%d",&t);
14     a=(y*t)/2;
15     printf("\nUcgenin Alani: %d 'dir",a);
16     return 0;
17 }
```

Ucgenin yukseklik degerini giriniz4

Ucgenin taban uzunlugu degerini giriniz6

Ucgenin Alani: 12 'dir

Process exited after 61.12 seconds with return value 0

Press any key to continue . . . _

Program Hataları

- Bug (Hata)
 - Programdaki hatalar
- Debugging (Hata Giderme)
 - Programdaki hataların düzeltilmesi
- Syntax (Sözdizimi) hataları
 - Programlama dili kurallarının ihlal edilmesi
 - Syntax hatalarını derleyici tespit eder. (Her zaman doğru satırı işaret etmeyebilir.)
- Run-time (Yürütme Aşaması) hataları
 - Program çalıştırıldığında bilgisayar tarafından tespit edilen hatalardır.
- Logic (mantık) hataları
 - Program algoritmasındaki hatalardır.
 - Genellikle tespiti zordur. Bilgisayar bu tip hataları bulamaz

Örnek

- Küçük harfi büyük harfe dönüştürme:
- ASCII tablosundan faydalanılabilir.
- Harf değerlerinin arasında sabit fark var.

Derleyen ASCII Chart (character codes 0 - 127)

064	@	080	P	096	`	112	p
065	A	081	Q	097	a	113	q
066	B	082	R	098	b	114	r
067	C	083	S	099	c	115	s
---	-	---	-	---	-	---	-

[*] ders3.3.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  /* run this program using the console pauser or
5     char harf;
6  int main() {
7
8     printf("\n Bir harf giriniz\n");
9     scanf("%c",&harf);
10    harf= harf-32;
11    printf ("\n Buyuk Harfiniz: %c",harf);
12
13    return 0;
14 }
15
16
```

C:\Users\Kemal\Desktop\C_Kemal_Ders\Kodlar\ders3.3.exe

Bir harf giriniz

R

Buyuk Harfiniz: R

Process exited after 4.825 seconds with return code 0
Press any key to continue . . .

077	M	093	j	109	m	125	y
078	N	094	^	110	n	126	~
079	O	095	_	111	o	127	û

Ödev

- Ödev 1:
Kullanıcıdan ekrana 4 adet tam sayı girmesini isteyin. Bu sayıların toplamını hesaplayıp, her bir sayının toplama göre yüzde değerini bulup ekrana yazdırın. Yüzdelik değerleri float türü değişken ile tutun ve bulduğunuz küsürlü 4 değerın toplamının 100 sayısını vermesi gerektiğini unutmayın