

**C++ Nedir ?**

Arttırılmış C demektir ve C dilinin geliştirilmiş halini yansıtır.  
bana göre lightC !

**C++ öğrenmeden önce herhangi bir programlama dilini öğrenmek gereklidir?**

Hayır, C++ basit ve açık bir dildir.

**Nesneye yönelik programlama nedir?**

Her program bileşeninin bir nesneden oluştuğunu göz önüne alan bir dildir. **NEDEN C++ ?**

- OOP (Object Oriented Programming-Nesneye yönelik programlama)
- Portatiflik
- Kısalık ve özlük
- Modülerlik
- Uyumluluk
- Hız

Bir programlama dilini öğrenmenin en iyi yolu bir program yazarak işe başlamaktır.

NOT : Programlama yaparken kodlarda kesinlikle TÜRKÇE karakter kullanılmaz.

(ş,ı,ö,ç,ü,ğ,İ,Ş,Ğ,Ö,Ü)

```
// C++ ile ilk programım
#include <iostream>
using namespace std;
int main ()
{ cout << "Selam dünya!";
  system("Pause");
  return 0;
}
```

**Merhaba Dünya!**  
Press any key to continue ...

```
// C++ ile ilk programım
```

(//) işaretlerinin sağındaki hiç birşey derleyici tarafından gözönüne alınmaz

```
#include <iostream>
```

(#) işareti ile başlayan satırlar ön-işlemci (derleyici) için yönlendirir.

```
using namespace std;
```

C++ kütüphanesinin tüm standart elemanlarının bildirimi

```
int main ()
```

Ana fonksiyonun tanımlandığı satır

```
int main (){
```

```
    cout << "Selam dünya!";
```

C++ söylem satırı. Bu satır (:) işareti ile sonlandırılıyor. "cout <<" → C++ standart çıkış

```
    system("Pause");
```

```
    return 0;
```

return, Ana fonksiyonun sonlandırılması.

```
}
```

```
int main ()
```

```
{ cout << "Selam dünya!";
```

```
  return 0;
```

```
}
```

```
int main (){ cout << "Selam dünya!";return 0;}
```

**ÖRNEK 1**

```
/* C++ ile ilk programım */
// C++ ile ilk programım

#include<iostream>
using namespace std;

int main ()
{
    cout<< "Merhaba Dünya!"<<sayi1<<"fatih marasli"<<sayi2<<endl;
    system("Pause");
    return 0;
}
```

**ÖRNEK 2**

```
// İkinci C++ programım
#include<iostream>
using namespace std;
int main ()
{
    int c;
    cout<<"Birtam sayı giriniz:";
    cin>> c;
    cout<<"Girilen sayı:"<<c<<endl;
    cout<< "Selam Dünya! ";
    cout<< "Ben bir C++ programıyım";
    return 0;
}
```

cout → bildirimi iostream dosyasında yapılır

---

**DEĞİŞKENLER VE DATA TÜRLERİ (VERİ TİPLERİ)**

Aklınızın bir köşesinde (belleginizde) "5" sayısını tutun

Aynı anda "2" sayısını da tutun

İlk tuttuğunuz sayının üzerine "1" ekleyin

1. bellek adresindeki değerden 2. bellek adresindeki değeri çıkartın ve sonucu aklınızda tutun

a = 5;

b = 2;

a = a + 1;

sonuc = a - b;

**TANIMLAYICILAR ( IDENTIFIERS)**

- Bir veya birden fazla karaktere sahip olabilirler ve tanımlayıcının boyu üzerine kısıtlama yoktur.

- Sadecealfabetikharfler, rakamlarve( \_ ) geçerlidir. (ASCII karakterseti)
- Tanımlayıcınınilk karakterialfabetikolmakzorundadır.**Bir tanımlayıcı hiç bir zaman “rakam” ile başlayamaz !**
- Tanımlayıcılarküçük/büyükharffarklarınaduyarlıdır**C/C++’da küçük ve BÜYÜK harfler farklı yorumlanır.**
- C/C++ anahtarsözcükleritanımlayıcıolarakkullanılamaz**  
asm, **auto**, **bool**, **break**, **case**, catch, **char**, **class**, **const**, const\_cast,continue, default, delete, **do**, **double**, dynamic\_cast, **else**, enum, explicit, export, extern, false, **float**, **for**, friend, **goto**, **if**, inline, **int**, **long**, mutable, **namespace**, **new**, operator, **private**, **protected**, **public**, register,reinterpret\_cast, **return**, **short**, **signed**, sizeof, static, static\_cast, struct, **switch**, template, this, throw, true, try, typedef, typeid, typename, union, **unsigned**, **using**, virtual, **void**, volatile, wchar\_t, **while**
- Bunlara ek olarak lojik tanımlayıcılar vardır.**  
and, and\_eq, bitand, bitor, compl, not, not\_eq, or, or\_eq, xor, xor\_eq

## DEĞİŞKENLERİN BİLDİRİMİ

Değişkenleri C++ da kullanabilmemiz için onların hangi **data türünde** olduğunun **bildirilmesi** gerekir.

DATA TÜRÜ	GEÇERLİ BİR DEĞİŞKEN TANIMLAYICI
int	a;
float	birsayi;

int a,b,c;  int a;  
int b;  
int c;

Tam sayı (integer) data türleri, temsil edilecek sayının büyüklüğüne göre, **char**, **short**, **long** ve işaretli (**signed**) veya işaretsiz (**unsigned**) olabilir.

**unsigned short int** KayıtlıOgrenciSayisi;

**signed int** BenimBankaHesabim;

**unsigned** bildirimi yapılmazsa **signed** olduğu varsayılır

**int** BenimBankaHesabim;

## Bir değişken genel (global) veya yerel (local) görünümlü olabilir.

**Genel değişkenler** tüm fonksiyonların dışında, kaynak kodunun ana fonksiyon kısmında bildirimleri yapılan değişkenlerdir

**Local değişkenler** ise bir fonksiyon veya bir blok içinde bildirimleri yapılan değişkenlerdir

```
#include <iostream>
using namespace std;
```

```
int Tamsayi;
char birKarakter;
char KarakterSicimi[20];
```

GENEL DEĞİŞKENLER

```
int main()
{
```

```
unsigned short yas;
float birSayi,DigerBirSayi;
```

YEREL DEĞİŞKENLER

```
cout << "Yasinizi Giriniz";
cin >> yas;
```

KOMUTLAR

```
...
...
}
```

## Değişkenlerin ilklendirilmesi/ Başlangıç değerleri atanması

Bildirimi yapılan bir yerel değişkenin ilk değeri belirsizdir. Bir değişken bildirimi yapıldığında değişkene bir başlangıç değeri verilmesi istenebilir.

```
tür tanımlayıcı = İlkDeğer;
```

```
tür tanımlayıcı (İlkDeğer);
```

```
int a = 1;
```

```
int a (1);
```

```
float PI= 3.14159265;
```

```
float PI(3.14159265);
```

..

Eğer bir karakter tanımlaması varsa değer ataması yapılırken tek tırnak içinde yapılır.

Char cinsiyet='e';

Eğer bir karakter dizisi yani kelime veya cümle ataması varsa çift tırnak içinde yapılır.

String adiniz="fatih";

```
stringbenimcumlem = "Bu bir string değer tutan değişkendir";
stringbenimcumlem ("Bu bir string değer tutan değişkendir ");
```

```
//Benim ilk stringim
#include<iostream>
#include<string>
usingnamespacestd;
intmain ()
{
stringbenimcumlem;
benimcumlem = "Bu benim ilk cumlemdir.";
cout<<benimcumlem<<endl;
return0;
}
```

## Karakter ve Karakter sicimi

'z'  
'p' "Merhaba Dünya"  
"Nasılsınız ?"

x → x tanımlayıcısı  
'x' → x karakter sabiti

## DERS02-ÖRNEK- 04

## KAÇIŞ(ESCAPE) KODLARI

\n	Yeni Satır(Newline)	\t	tab	\f	Yeni sayfa
\r	taşıyıcıyı döndür (Carrige return)	\b	Bir geri git	\v	Yatay tab

\a →alert yani alarm sesi

Soru işaret ve çift tırnağı derleyici aslında kod parçası olarak göreceği için doğrudan ekran çıktısında göremezsiniz. Görmek istediğinizde ona uygun kodlar mevcuttur.

\? →ekrana soru işareti çıktısı verir.

\” → ekrana çift tırnak çıktısı verir.

\' → ekrana tek tırnak çıktısı verir.

```
75          // decimal
0113        // octal
0x4b        // hexadecimal
```

```
75          // int
75u         // unsigned int
75l         // long
75ul        // unsigned long
```

```
3.14159     // 3.14159
6.02e23     // 6.02 x 10^23
1.6e-19     // 1.6 x 10^-19
3.0         // 3.0
```

```
3.14159L    // long double
6.02e23f    // float
```

Değer atamalarında üstteki gibi harflendirmelerle karşılaşabilirsiniz. Bunlar atanan değerın tipini belirtmek için kullanılır.

### Tanımlanmış Sabitler - defined constants (#define)

**#define** tanımlayıcı değeri ;

```
#define PI 3.14159265
#define YENISATIR '\n'
```

#define bir C++ söylemi değil, ön-işlemler için bir yönlendiricidir.

Bundan dolayı satır sonunda (;) bulunmaz



### Bildirimli sabitler - Declared constants (const)

```
const int tampuan    = 100 ;
const char tabulator = '\t' ;
const char backspace = '\b' ;
```

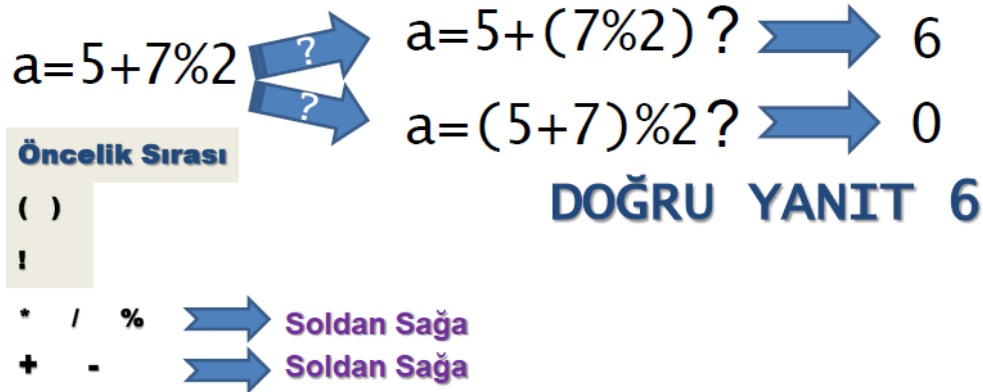
### ÖRNEK

```
// tanımlanmış sabitler
#include<iostream>
using namespace std;
#define PI 3.14159
#define yenisatir '\n'
int main ()
{
    double yaricap=5.0;
    double cevre;
    cevre = 2 * PI * yaricap;
    cout<<cevre;
    cout<<yenisatir;
    return 0;
}
```

-----

## OPERATÖRLER

### OPERATÖRLERİN ÖNCELİK SIRALARI



Operatörlerde öncelik sırası şu şekildedir. Birden fazla işlem varsa bir kod satırında önce ÇARPMA sonra BÖLME sonra YÜZDE(MOD lu bölme işlemi yani bölümünden kalan hesabı) sonra TOPLAMA ve en son ÇIKARMA işlemi yapılır.

$A = 11\%3$ ; kod satırı sonucunda A değeri 2 olur. Çünkü 11'in 3'e bölümünden kalan 2'dir.

Bazı aritmetiksel işlemlerde kısaltmalar kullanılır.

Toplam= Toplam + sayac şeklinde tanımladığımız kod satırındaki amaç Toplamın bir önceki değeri ile sayac değerini topluyoruz ve yine Toplam değişkenine atıyoruz. Bu kod satırını daha kısa yöntemle yazarsak Toplam += sayac şeklinde de yazabiliriz.

Daha fazla örnek verirsek  $a = a - 2$  yerine  $a -= 2$  yazabiliriz.

$X = X * 5$  yerine  $X *= 5$  yazabiliriz.

Bunların dışında birde tek tek artırma ve azaltma değerleri var.

Örneğin ;

$A = 4$  diyelim ve bu A değişkeni for döngüsü içinde kullanılmış olsun.

For (  $A=4$  ;  $A<10$  ;  $A++$  )

{

.... Belli komutlar kullanıldığını varsayalım

}

Burada döngü ilk başta  $A=4$  için bir sefer çalışır yani for' un {..} küme parantezi arasında yazılan komutlar bir sefer çalışır. Sonra  $A++$  komutu A sayısını 1 artırır. Bu demek oluyor ki önce for'un içindeki işlemler yapıldıktan sonra A 1 artar. Ama  $++A$  demiş olsaydık



for içindeki komutlar çalışmadan önce A 1 artar sonra for içindekiler çalışır. Burada ++ ifadesinin önce ya da sonra yazılmasındaki fark budur.

Bunların dışındaki operatörlerden birkaçı da şunlardır :

== eşittir

!= eşit değildir

< küçüktür

> büyüktür

<= küçük eşittir

>= büyük eşittir Genelde bu operatörler if koşullarında kullanılır.

## Lojik Operatörler

! değil(NOT)

&& ve (AND)

|| veya (OR)

## Bitsel Operatörler

operator	asm equivalent	description
&	AND	Bitwise AND
	OR	Bitwise Inclusive OR
^	XOR	Bitwise Exclusive OR
~	NOT	Unary complement (bit inversion)
<<	SHL	Shift Left
>>	SHR	Shift Right

## Koşullu Operatörler

İntFth = 7==5 ? 4 : 3

StringFth = vize==60 ? "geçti" : "kaldı" // 7 5+2'ye eşitse 4 döner değilse 3 döner.

Fth = 5>3 ? a : b // 5 3'ten büyükse sonuç olarak a döner değilse b döner.

Buyukdeger = a>b ? a : b // a b'den büyükse sonuç olarak a döner değilse b döner

## Açık Tip Dönüşüm Operatörü

inti=2;

floatf = 3.14;

f = (float) i;

Buradaki komut satırlarında ilk başta i değişkeni integer olarak tanımlanıyor. Sonra float tipinde f değişkeni tanımlanıyor ve f değişkenine 3.14 değeri atanıyor. Son satırda ise f değişkeni INTEGER tipine dönüştürülüyor ve i değişkenine atanıyor değeri. Yani f nin tuttuğu 3.14 değeri integer e dönüştürülüp i değişkenine atanıyor. 3.14 değeri float tan integer a dönüşünce küsürlü kısmını kaybeder ve sadece 3 değerine dönüşür ve bu da i değişkenine atanmış olur. Sonuç olarak i = 3 tür.

-----

## Temel Giriş/Çıkış Komutları

### Standart çıkış (cout)

```
#include<iostream>
usingnamespacestd;
cout<< " bir cikticumlesi"; //Ekрана bir cikti cümlesi yazar
cout<< 120; // Ekрана 120 yazar
intx = 120;
cout<< x; //Ekрана x değişkeninin tuttuğu değeri yazar
conststring SELAM= "FATİH";
cout<< "SELAM ";//Ekрана SELAM yazar
cout<< SELAM<<endl; // Ekрана SELAM değişkeninin tuttuğu değeri yazar.Yani
FATİH yazar.
cout<< SELAM << " Bu satir"<<"birC++ satiri"<<endl;//Ekрана SELAM değişkeninin
tuttuğu değeri sonra “Bu satir bir C++ satiri” cümlesini yazıp aşağı satıra geçer
cout<< SELAM << " Bu satir"<<"birC++ satiri\n";
```

### Standart giriş (cin)

```
#include<iostream>
usingnamespacestd;
int sayi1,sayi2;
cout<<"İkisayigiriniz:";
cin>> sayi1>>sayi2;//aynı anda iki değeri birden okutmuş oluyoruz. cin>> sayi1; cin>>sayi2;
cout<<"ikisayitoplami= "<<sayi1+sayi2;
```

### cin ve karakter sicimleri (strings)

```
#include<iostream>
#include<string>
usingnamespacestd;
stringname,takim;
cout<<"AdiniziGiriniz: ";
getline(cin,name);// getline -> Bir satir al
cout<<"Selamsize "<<name<<"\n";
cout<<"Hangifutboltakiminidestekliyorsunuz?: ";
getline(cin,takim);
cout<<takim<<" ?"<<endl;
```

## stringstream

```
#include<iostream>
#include<string>
#include<sstream>
usingnamespacestd;
stringbirsicim;
float Fiyat=0;
intAdedi=0;
cout<< "Fiyatgiriniz: “;
getline(cin,birsicim);
stringstream(birsicim) >>Fiyat;
cout<< "Adedi: “;getline(cin,birsicim);
stringstream(birsicim) >>Adedi;
cout<< "Tutari: " << Fiyat*Adedi<< " TL\n";
```

---

## KOŞULLU YAPI (IF/ELSE if/else)

If ( eğer ) komutundaki koşul gerçekleşiyorsa if e dahil olan alt satırındaki komutlar çalışır. Ve if 'in altındaki komutlardan hangisi if e göre çalışacaksa o {} parantezleri arasına alınır.

```
if(x==100)
{
cout<< "x=100“;
cout<< x;
}
```

Eğer x'in tuttuğu değer 100 e eşitse ekrana önce x=100 yazdıracak sonra da x değişkeninin tuttuğu değeri yazacak.

```
if(x > 0)
cout<< "x > 0 ‘tür”;
```

```
elseif(x < 0)
cout<< "x < 0”;
```

```
else
cout<< "x =0”;
```

Üstteki örnekte çoklu if yapısı. Eğer birden fazla şart ya da ihtimal varsa else if diyerek devam ettirilir. Son satır else ile bitirilir.

```
cout<<"Bir sayi giriniz:";
cin>>n;
if(n%2==0)   cout<<"n cift \n";
else        cout<<"n tek \n";
```

Üstteki örnekte Bir sayı giriniz diye ekrana yazı çıkar. Biz klavyeden bir sayı tuşlarız ve tuşladığımız değeri program cin komutu ile okuyarak n değişkeninde tutar. Eğer bu n değişkeninin 2 ye bölümünden kalan 0 ise ekrana çift yazdır ve \n ile bir satır aşağı geçmesi sağlanır. Else yani önceki şart sağlanmazsa ekrana tek yazdıracak ve yine \n ile bir satır aşağı geçmesi sağlanacak.

```
if(x > 0)
{
    cout<< " x";
    cout<< " pozitif"<<endl;
}
elseif(x < 0)
{
    cout<< " x ";
    cout<< " negatif"<<endl;
}
else
{
    cout<< "x sıfır"<<endl;
}
```

## TEKRARLAMA YAPILARI (DÖNGÜLER)

### While Döngüsü

```
#include<iostream>
usingnamespacestd;
intmain ()
{
    intn = 0;
    cout<<"Baslangicsayisini giriniz> ";
    cin>> n;
    while(n>0)
    {
        cout<< n <<" ";
        --n;
    }
    cout<<"ATESSS!\n";
    return0;
}
```

#### EKRAN ÇIKTISI

Baslangicsayisini giriniz > 8  
8, 7, 6, 5, 4, 3, 2, 1, ATESSS!

### Do-While Döngüsü

```
#include<iostream>
usingnamespacestd;
intmain ()
{
```

```
unsigned long n;  
do{  
    cout<<"Sayi giriniz (0 dan büyük): ";  
    cin>> n;  
    cout<<"Girdiginiz sayi: " << n <<"\n";  
}while(n != 0);  
return 0;  
}
```

## For Döngüsü

```
#include<iostream>  
using namespace std;  
int main ()  
{  
  
    for(int n=10; n>0; n--)  
    {  
        cout<< n <<" ";  
    }  
  
    cout<<"ATESSS!\n";  
    return 0;  
}
```

EKRAN ÇIKTISI

10, 9, 8, 7, 6, 5, 4, 3, 2, 1, ATESSS!

```
for( n=0, i=100 ; n!=i ; n++, i-- )  
{  
    // whatever here...  
}
```

## ATLAMA DURUMLARI

### Break Komutu

```
#include<iostream>  
using namespace std;  
int main ()  
{  
    int n;  
    for(n=10; n>0; n--)  
    {  
        cout<< n <<" ";  
        if(n==3)  
        {  
            cout<<"Asagi sayma iptal edildi!";  
            break;  
        }  
    }  
}
```

```
    }  
}  
  
return 0;  
}
```

EKRAN ÇIKTISI

10, 9, 8, 7, 6, 5, 4, 3, Asagi sayma iptal edildi!

## Devam Durumu

```
#include<iostream>  
using namespace std;  
int main ()  
{  
    for(int n=10; n>0; n--)  
    {  
        if(n==5)  
        {  
            continue;  
        }  
        cout<< n <<" ";  
    }  
    cout<<"ATESS!\n";  
    return 0;  
}
```

EKRAN ÇIKTISI

10, 9, 8, 7, 6, 4, 3, 2, 1, ATESS!

## GoTo Durumu

```
#include<iostream>  
using namespace std;  
int main ()  
{  
    int n=10;  
    fatih:  
    cout<< n <<" ";  
    n--;  
    if(n>0)  
    {  
        goto fatih;  
    }  
    cout<<"ATESS!\n";  
    return 0;  
}
```

}

**EKRAN ÇIKTISI**

10, 9, 8, 7, 6, 5, 4, 3, 2, 1, ATESS!

**Seçici Yapı Switch/Case****switch (x)**

```
{
case 1:cout<< "x : 1"; break;
case 2:cout<< "x : 2"; break;
default:cout<< " x degeribilinmiyor";
}
```

Switch komutunun açılımına bakarsak switch(x) burada değişkenimiz x .case 1 deki amaç şudur. Switch ile tanımlanan x değişkeni case 1 yani durum 1 yani x=1 ise ekrana x:1 yazdırır.

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	<b>NUL</b> (null)	32	20	040	&#32;	Space	64	40	100	&#64;	@	96	60	140	&#96;	`
1	1	001	<b>SOH</b> (start of heading)	33	21	041	&#33;	!	65	41	101	&#65;	A	97	61	141	&#97;	a
2	2	002	<b>STX</b> (start of text)	34	22	042	&#34;	"	66	42	102	&#66;	B	98	62	142	&#98;	b
3	3	003	<b>ETX</b> (end of text)	35	23	043	&#35;	#	67	43	103	&#67;	C	99	63	143	&#99;	c
4	4	004	<b>EOT</b> (end of transmission)	36	24	044	&#36;	\$	68	44	104	&#68;	D	100	64	144	&#100;	d
5	5	005	<b>ENQ</b> (enquiry)	37	25	045	&#37;	%	69	45	105	&#69;	E	101	65	145	&#101;	e
6	6	006	<b>ACK</b> (acknowledge)	38	26	046	&#38;	&	70	46	106	&#70;	F	102	66	146	&#102;	f
7	7	007	<b>BEL</b> (bell)	39	27	047	&#39;	'	71	47	107	&#71;	G	103	67	147	&#103;	g
8	8	010	<b>BS</b> (backspace)	40	28	050	&#40;	(	72	48	110	&#72;	H	104	68	150	&#104;	h
9	9	011	<b>TAB</b> (horizontal tab)	41	29	051	&#41;	)	73	49	111	&#73;	I	105	69	151	&#105;	i
10	A	012	<b>LF</b> (NL line feed, new line)	42	2A	052	&#42;	*	74	4A	112	&#74;	J	106	6A	152	&#106;	j
11	B	013	<b>VT</b> (vertical tab)	43	2B	053	&#43;	+	75	4B	113	&#75;	K	107	6B	153	&#107;	k
12	C	014	<b>FF</b> (NP form feed, new page)	44	2C	054	&#44;	,	76	4C	114	&#76;	L	108	6C	154	&#108;	l
13	D	015	<b>CR</b> (carriage return)	45	2D	055	&#45;	-	77	4D	115	&#77;	M	109	6D	155	&#109;	m
14	E	016	<b>SO</b> (shift out)	46	2E	056	&#46;	.	78	4E	116	&#78;	N	110	6E	156	&#110;	n
15	F	017	<b>SI</b> (shift in)	47	2F	057	&#47;	/	79	4F	117	&#79;	O	111	6F	157	&#111;	o
16	10	020	<b>DLE</b> (data link escape)	48	30	060	&#48;	0	80	50	120	&#80;	P	112	70	160	&#112;	p
17	11	021	<b>DC1</b> (device control 1)	49	31	061	&#49;	1	81	51	121	&#81;	Q	113	71	161	&#113;	q
18	12	022	<b>DC2</b> (device control 2)	50	32	062	&#50;	2	82	52	122	&#82;	R	114	72	162	&#114;	r
19	13	023	<b>DC3</b> (device control 3)	51	33	063	&#51;	3	83	53	123	&#83;	S	115	73	163	&#115;	s
20	14	024	<b>DC4</b> (device control 4)	52	34	064	&#52;	4	84	54	124	&#84;	T	116	74	164	&#116;	t
21	15	025	<b>NAK</b> (negative acknowledge)	53	35	065	&#53;	5	85	55	125	&#85;	U	117	75	165	&#117;	u
22	16	026	<b>SYN</b> (synchronous idle)	54	36	066	&#54;	6	86	56	126	&#86;	V	118	76	166	&#118;	v
23	17	027	<b>ETB</b> (end of trans. block)	55	37	067	&#55;	7	87	57	127	&#87;	W	119	77	167	&#119;	w
24	18	030	<b>CAN</b> (cancel)	56	38	070	&#56;	8	88	58	130	&#88;	X	120	78	170	&#120;	x
25	19	031	<b>EM</b> (end of medium)	57	39	071	&#57;	9	89	59	131	&#89;	Y	121	79	171	&#121;	y
26	1A	032	<b>SUB</b> (substitute)	58	3A	072	&#58;	:	90	5A	132	&#90;	Z	122	7A	172	&#122;	z
27	1B	033	<b>ESC</b> (escape)	59	3B	073	&#59;	;	91	5B	133	&#91;	[	123	7B	173	&#123;	{
28	1C	034	<b>FS</b> (file separator)	60	3C	074	&#60;	<	92	5C	134	&#92;	\	124	7C	174	&#124;	
29	1D	035	<b>GS</b> (group separator)	61	3D	075	&#61;	=	93	5D	135	&#93;	]	125	7D	175	&#125;	}
30	1E	036	<b>RS</b> (record separator)	62	3E	076	&#62;	>	94	5E	136	&#94;	^	126	7E	176	&#126;	~
31	1F	037	<b>US</b> (unit separator)	63	3F	077	&#63;	?	95	5F	137	&#95;	_	127	7F	177	&#127;	DEL

## BİLEŞİK DATA TIPLERİ - DİZİNLER

Eğer  $n$  adet elemana sahip bir dizin'in adı  $a$  ise:

$a_0, a_1, a_2, \dots, a_{n-1}$   $a_0$  : dizinin ilk elemanı  
 $a_{n-1}$  : dizinin son elemanı

C/C++ da Dizilerin ilk indisi "0" dan başlar  
 (Fortran da "1" !)

$a[0], a[1], a[2], \dots, a[n-1]$  Bu temsil şekli her dizin elemanının ilk  $a[0]$  elemanından kaç adım ötede olduğunu bildirir.

float a[5];

a[0]	0	11.11
a[1]	1	33.33
a[2]	2	55.55
a[3]	3	77.77
a[4]	4	99.99

örnek:  
 $a[3]$  ,  $a[0]$  elemanından 3 adım ötededir.

[Ders07-01](#)

```
int main(){
    const int N = 5;
    double a[N];
    cout << N << " adet sayi giriniz:\t"<<endl;
    for (int i=0; i<N; i++){
        cout <<" a["<<i<<"]=";
        cin >> a[i];
    }
    cout << "Ters siralanmis olarak: ";
    for (int i=N-1; i>=0; i--)
        cout << "\t" << a[i];
}
```

[Ders07-02](#)

sizeof(a) → büyüklük (byte)

```
int main(){
    float a[] = { 22.2,44.4, 66.6 };
    cout << sizeof(a) << endl;
    cout << sizeof(float) << endl;
    int boy = sizeof(a)/sizeof(float);
    for (int i=0; i<boy; i++)
        cout << "\ta[" << i << "] = " << a[i] << endl;
}
```

12 byte

4 byte

3

[Ders07-03](#)



## DİZİNLERE BAŞLANGIÇ DEĞERLERİ ATANMASI

```
int main()
{ const int N=4;
float a[N] ;
for (int i=0; i<N; i++)
cout << "\ta[" << i << "] = " << a[i] << endl;
float a[N] = { 33.3,44.4,55.5,66.6 };
for (int i=0; i<7; i++)
cout << "\ta[" << i << "] = " << a[i] << endl;
```

```
a[0] = 7.93931e+033
a[1] = 2.89037e-040
a[2] = 2.8888e-040
a[3] = 1.12104e-044
```

[Ders07-04](#)

[Ders07-05](#)

```
a[0] = 33.3
a[1] = 44.4
a[2] = 55.5
a[3] = 66.6
a[4] = 7.93782e+033
a[5] = 5.34659e+036
a[6] = 5.31691e+036
```

```
const int boy=3;
float a[boy] = { 22.2,44.4,66.6 };
float x=11.1;
cout << "x = " << x << endl;
cout << "a[2] = " << a[2] << endl;
cout << "a[3] = " << a[3] << endl;
cout << "a[4] = " << a[4] << endl; a[4] = 44.4;
cout << "a[4] = " << a[4] << endl;
cout << "a[5] = " << a[5] << endl; a[5] = 55.5;
cout << "a[5] = " << a[5] << endl;
```

[Ders07-06](#)

```
int a[]={22,44,66,88 };
cout << "a = " << a;
// a[0] rin adresi
a = 0x23ff60
```

[Ders07-07](#)

```
x = 11.1
a[2] = 66.6
a[3] = 1.12104e-044
a[4] = 7.93782e+033
a[4] = 44.4
a[5] = 5.34659e+036
a[5] = 55.5
```

**Çok-boyutlu dizinler**

```

#define SUTUN 5
#define SATIR 3
int matris [SATIR] [SUTUN];
int n,m;
int main (){
for(n=0;n<SATIR;n++){
    for(m=0;m<SUTUN;m++){
        matris [n] [m]=(n*SUTUN)+m;
        cout<<'['<< n << m << ']'<< setw(3)<< matris[n][m]<< ' ';
    }
    cout << endl;
}
}

```

**matris [1][3]**

**program çıktısı:**

```

[00] 0 [01] 1 [02] 2 [03] 3 [04] 4
[10] 5 [11] 6 [12] 7 [13] 8 [14] 9
[20] 10 [21] 11 [22] 12 [23] 13 [24] 14

```

[Ders07-01](#)**FONKSİYONLAR (I)****STANDART C++ KÜTÜPHANE FONKSİYONLARI**

Function	Description	Example
acos(x)	inverse cosine of x (in radians)	acos(0.2) returns 1.36944
asin(x)	inverse sine of x (in radians)	asin(0.2) returns 0.201358
atan(x)	inverse tangent of x (in radians)	atan(0.2) returns 0.197396
ceil(x)	ceiling of x (rounds up)	ceil(3.141593) returns 4.0
cos(x)	cosine of x (in radians)	cos(2) returns -0.416147
exp(x)	exponential of x (base e)	exp(2) returns 7.38906
fabs(x)	absolute value of x	fabs(-2) returns 2.0
floor(x)	floor of x (rounds down)	floor(3.141593) returns 3.0
log(x)	natural logarithm of x (base e)	log(2) returns 0.693147
log10(x)	common logarithm of x (base 10)	log10(2) returns 0.30103
pow(x,p)	x to the power p	pow(2,3) returns 8.0
sin(x)	sine of x (in radians)	sin(2) returns 0.909297
sqrt(x)	square root of x	sqrt(2) returns 1.41421
tan(x)	tangent of x (in radians)	tan(2) returns -0.917347

Header File	Description
<cassert>	Defines the assert() function
<ctype>	Defines functions to test characters
<cfloat>	Defines constants relevant to floats
<climits>	Defines the integer limits on your local system
<cmath>	Defines mathematical functions
<cstdio>	Defines functions for standard input and output
<cstdlib>	Defines utility functions
<cstring>	Defines functions for processing strings
<ctime>	Defines time and date functions

[Ders07-fonktest](#)

## Kare kök Fonksiyonu *sqrt()*

```
#include <cmath> // sqrt() fonksiyonunu tanımlar
#include <iostream> // cout nesnesini tanımlar

using namespace std;
int main()
{ // sqrt() fonksiyonu testi
  for (int x=0; x < 6; x++)
    cout << "\t" << x << "\t" << sqrt(x) << endl;
}
```

### Ders07-03

0	0
1	1
2	1.41421
3	1.73205
4	2
5	2.23607

## KULLANICI-TANIMLI FONKSİYONLAR

```
type name ( parameter1, parameter2, ...) { statements }
```

```
tür fonksiyon_adı(parametre1, parametre2,...){ söylemler}
```

- tür - fonksiyon tarafından döndürülecek olan data türünün özdeşleyicisi
- fonksiyon\_adı - fonksiyonun çağırılmasını sağlayan fonksiyon adı özdeşleyicisi
- parametre(ler) - her parametre bir data türünü ve onu izleyen bir değişken özdeşleyicisini içerir. (örnek: int x)
- söylemler - fonksiyonun { } arasında kalan söylemler bloğu

```
int kub (int x) int main (){
{
  int z = kub(3);
  return x*x*x; cout<<"Sonuc = " << z;return 0;}
}
```

### Ders07-05

```
#include <iostream>
#include <conio.h>
#include <iomanip>;
```

```
int glb_X = 0;
```

Global Değişken

```
int main ()
{
    int SiraNo = 0;
    do{
        cout << " Sira#" << setw(2)<< ++SiraNo;
        cout << " glb_X " << setw(2)<< glb_X++ << endl;
        getch();
    }while(glb_X<21);
    getch();
    return 0;
}
```

```
#include <iostream>
#include <conio.h>
#include <iomanip>
using namespace std;
```

```
long faktorial(int n)
{ // n! = n*(n-1)*(n-2)*...*(2)*(1) dondurur
  if (n < 0) return 0;
  int f=1;
  while (n > 1)
  f *= n--;
  return f;
}
```

```
-1 0
0 1
1 1
2 2
3 6
4 24
5 120
```

```
int main()
{ // faktorial() fonksiyon testi
  for (int i=-1; i<6; i++)
    cout<<setw(3)<<i<<" " <<setw(3)<<faktorial(i)<< endl;
}
```

## Herhangi bir türde olmayan fonksiyonlar-void kullanımı.

```
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;
```

[Ders07-09](#)

```
void printmesaj ()
{
    cout << " Bu Benim türü olmayan(tipsiz) fonksiyonum!";
}

int main ()
{
    printmesaj ();
    getch();
    return 0;
}
```

## FONKSİYONLAR

```
#include <iostream> //
#include <conio.h>
#include <iostream>
#include <iomanip>
using namespace std;
```

& ↔ Değer yerine referans yollarır

void ikikat(int & a, int & b, int & c)

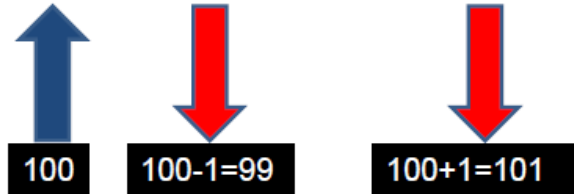
```
void ikikat(int & a, int & b, int & c){
    a*=2;
    b*=2;
    c*=2;
}
```

ikikat (x, y, z);

```
int main ()
{
    int x=1, y=3, z=7;
    ikikat (x, y, z);
    cout << "x=" << x << ", y=" << y << ", z=" << z << endl;}
```

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

```
void oncekisonraki (int x, int& onceki, int& sonraki)
{
    onceki = x-1;
    sonraki = x+1;
}
```



```
int main ()
{
    int x=100, y, z;
    oncekisonraki (x, y, z);
    cout << "Onceki =" << y << endl;
    cout << "Sonraki=" << z << endl;
    cout << "x=" << x << endl;
    return 0;
}
```

### ***Varsayılan Değerlerin (default) Parametre olarak kullanımı***

//Fonksiyonlarda varsayılan parametre kullanımı

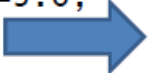
```
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;


int bolum (int a, int b=2)
{
    int r;
    r=a/b;
    return (r);
}
```

```
int main ()
{
    cout << bolum (12);      ➡      6
    cout << endl;
    cout << bolum (20,4);      ➡      5
    getch();
}
```

## Aşırı yüklenim - overloading

C++ da farklı iki fonksiyon eğer parametrelerinin **sayıları** veya **data türleri farklı** ise aynı fonksiyon\_adına sahip olabilirler

```
// asiri yüklenmiş fonksiyon
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;
int carpim (int a, int b){
    return (a*b);}
float carpim (float a, float b, float c){
    return (a*b*c);}
int main (){
    int x=5,y=2;
    float fk=5.0,fl=2.0,fm=3.0;
    cout << carpim (x,y);  10

    cout << "\n";cout<< carpim (fk,fl,fm);  30
    cout << "\n";
    return 0;
```

## Yinelemeli Fonksiyon - kendini çağırma özelliği olan fonksiyonlar

$$(n!) = n * (n-1) * (n-2) * (n-3) \dots * 1$$

```
long faktoriyal (long a)
{
    if (a > 1)
        return (a * faktoriyal (a-1));
    else
        return (1);
}
int main ()
{
    long sayi;
    do{
        cout << "Bir sayi giriniz: ";
        cin >> sayi;
    }while (sayi >= 32);
    cout << sayi << "! = " << faktoriyal (sayi);
    return 0;
}
```

[Ders08-05a](#)

[Ders08-05](#)



### ***Fonksiyon Bildirimleri***

```
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;
void tek (int a);
void cift (int a);
int main (){
    int i;
    do {
        cout << "Bir sayi giriniz (cikis 0): ";
        cin >> i;
        tek (i);
    } while (i!=0);
    return 0;}

void tek (int a){
    if ((a%2)!=0) cout << "Tek sayi.\n"; else cift (a);
}
void cift (int a){
    if ((a%2)==0) cout << "Cift sayi.\n"; else tek (a);
}
```

[Ders08-06](#)

### ***Parametrelerin Dizinler Olması Durumu***

```
#include <cmath>
#include <iostream>
#include <conio.h>
using namespace std;
void DizinYazdir (int arg[], int uzunluk)
{
    for (int n=0; n<uzunluk; n++)
        cout << arg[n] << " ";
        cout << "\n";
}

int main (){
    int Dizin[] = {5,10,15};
    Dizin[1]++; Dizin[2]++;
    DizinYazdir(Dizin,3);
    getch();
    return 0;
}
```



5 11 16



```
#include <conio.h>
using namespace std;
int main(){
    int m,n;
    do
    {
        cout << "iki sayi giriniz"<<endl;
        cout << "1. sayi : "; cin >> m;
        cout << "2. sayi : "; cin >> n;
        cout << "\tmax(" << m << "," << n << ") = " << max(m,n) << endl;
    }
    while (m != 0);
}
```

```
// verilen iki tam sayıdan büyük olanini
döndürür:
int max(int x,int y)
{ if (x < y) return y;
  else return x;
}
```

### **Permütasyon Fonksiyonu**

```
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;

long fakt(int n){
    if (n < 0) return 0;
    int f = 1;
    while (n > 1) f *= n--;
    return f;
}

long perm(int n,int k){
    if (n < 0 || k < 0 || k > n)
        return 0;
    return fakt(n)/fakt(n-k);
}

int main(){
    for (int i = -1; i < 8; i++){ for (int j=-1; j <= i+1; j++)
        cout << " " << perm(i,j);
        cout << endl;}}
}
```

$$P(n, k) = \frac{n!}{(n-k)!}$$

$$P(5, 2) = \frac{5!}{(5-2)!} = \frac{5!}{3!} = \frac{120}{6} = 20$$

```
0 0
0 1 0
0 1 1 0
0 1 2 2 0
0 1 3 6 6 0
0 1 4 12 24 24 0
0 1 5 20 60 120 120 0
0 1 6 30 120 360 720 720 0
0 1 7 42 210 840 2520 5040 5040 0
```

[Ders08-09](#)

```
#include <cmath> //
#include <iostream> //
#include <conio.h>
using namespace std;

void degistokus(float& x,float& y)
{ // x ve y nin degerlerini degis-
  tokus eder:
  float gecici = x;
    x = y;
    y = gecici;
}

int main()
{
float a = 22.2,b = 44.4;
cout << "a = " << a << ", b = " << b << endl;

degistokus(a,b);

cout << "a = " << a << ", b = " << b << endl;
}
```

## GÖSTERGEÇLER ve REFERANSLAR (Pointers and References)

Bilgisayar belleği baytlardan oluşan çok uzun bir dizin olarak düşünülebilir.

256 MB lık bir RAM (256 megabytes of random-access memory)

aslında 268,435,456 byte içerir.

$$256 \times 1024 \times 1024 = 268,435,456 \Leftrightarrow 2^{28}$$

256 MB bilgisayarın bellek adresleme aralığı:


0 - 268,435,455 desimal


0x00000000 - 0xffffffff hexadecimal

Bir değişken bildirimi 3 temel öğeye sahiptir:

- *değişkenin adı*
- *türü*
- *bellek adresi*

*int n;*      *değişkenin adı : n*  
                  *değişkenin türü : int*  
                  *değişkenin adresi : 0x0064fdf0 (varsayılm)*

0x0064fdf0  
*n*  *int*      *int n; //(4byte)*

0x0064fdf0  
*n*  *int*      *int n = 44;*

0x0064fdee	
0x0064fdef	
0x0064fdf0	
0x0064fdf1	
0x0064fdf2	
0x0064fdf3	
0x0064fdf4	
0x0064fdf5	
0x0064fdf6	

0x00000000  
 0x00000001  
 0x00000002  
 0x00000003  
 0x00000004  
 0x00000005  
 0x00000006  
 0x00000007  
 0x00000008

0x0064fdee  
 0x0064fdef  
 0x0064fdf0  
 0x0064fdf1  
 0x0064fdf2  
 0x0064fdf3  
 0x0064fdf4  
 0x0064fdf5  
 0x0064fdf6  
 0x0064fdf7  
 0x0064fdf8  
 0x0064fdf9  
 0x0064fdfa  
 0x0064fdfb  
 0x0064fdfc  
 0x0064fdfd  
 0x0064fdfe  
 0x0064fdff  
 0x0064fe00  
 0x0064fe01

0xffffffff7  
 0xffffffff8  
 0xffffffff9  
 0xffffffffffa  
 0xffffffffffb  
 0xffffffffffc  
 0xffffffffffd  
 0xffffffffffe  
 0xffffffffff

C++ da, **değişken adresi & referans operatörü** kullanılarak elde edilir (**Adres Operatörü**)

**&n** ifadesi değişkeninin bellekdeki adresini elde eder

```
int main()
{ int n=44;
cout << "n = " << n << endl; // n değerini yaz
cout << "&n = " << &n << endl; // n'in adresini yaz
}
```

Ders09-01

```
n = 44
&n = 0x22ff74
```

**REFERANSLAR** - Bir başka **değişken için diğer bir takma addır (namı-diğer!)** *tür& ref-adı = değ-adı;*

Ders09-02

```
int main()
{ int n=44;
int& rn=n; // rn, n ile eşanlamlı
cout << "n = " << n << ", rn = " << rn << endl; --n;
cout << "n = " << n << ", rn = " << rn << endl; rn *= 2;
cout << "n = " << n << ", rn = " << rn << endl;
}
```

**Referanslar ayrıık değişkenler değildir**

```
int main()
{ int n=44;
int& rn=n; // rn, n ile eşanlamlı
cout << " &n = " << &n << ", & rn = " << &rn << endl;
int& rn2=n; // rn2, n için bir başka eşanlamlı
int& rn3=rn; // rn3, n için bir başka eşanlamlı
cout << "&rn2 = " << &rn2 << ",&rn3 = " << &rn3 << endl;
}
```

**n ve rn aynı adrese sahipler:**

**0x22ff74**

**Aynı nesne için sadece farklı adlar**

**Bir nesne birkaç referansa sahip olabilir**

**rn, rn2 ve rn3 hepsi aynı nesneye referanslardır**

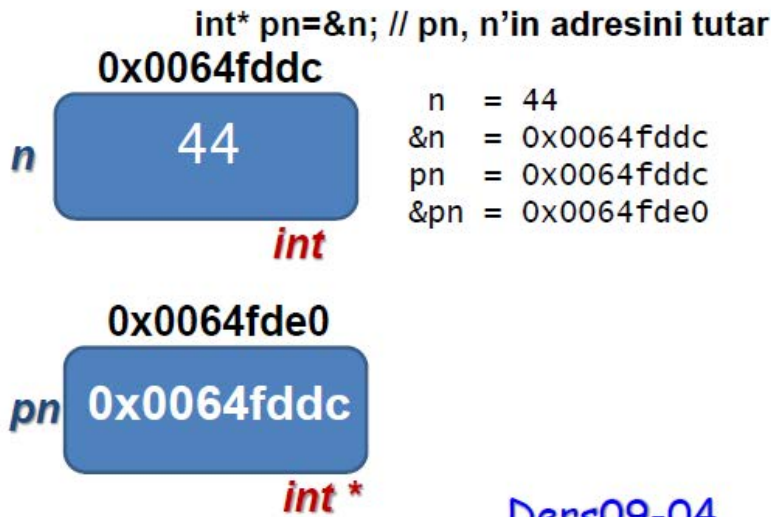
```
&n = 0x22ff74 , &rn = 0x22ff74
```

```
&rn2 = 0x22ff74 , &rn3 = 0x22ff74
```

## GÖSTERGEÇLER - POINTERS

*Referans operatorörü & bir değişken üzerine uygulandığında değişkenin bellek adresini döndürür*

*Bu adres başka bir değişken aracılığıyla da saklanabilir. Adresin saklandığı değişken türüne **göstergeç (pointer)** denir.*



0x0064fdda	
0x0064fddb	
0x0064fddc	44
0x0064fddd	
0x0064fdde	
0x0064fddf	
0x0064fde0	0x0064fddc
0x0064fde1	
0x0064fde2	
0x0064fde3	
0x0064fde4	
0x0064fde5	

Ders09-04

```
int main ()
{
    int ilkdeger, ikincideger;
    int * benim_ptr;
    benim_ptr = &ilkdeger;
    *benim_ptr = 10;
    benim_ptr = &ikincideger;
    *benim_ptr = 20;
    cout << " &ilkdeger = " << &ilkdeger << " ilkdeger = " << ilkdeger << endl;
    cout << " &ikinci deger = " << &ikincideger << " ikinci deger = " << ikincideger << endl;
    return 0;
}
```

```
int main ()
{
    int ilkdeger = 5, ikincideger = 15;
    int * p1, * p2;

    p1 = &ilkdeger; // p1 = ilkdeger'in adresi
    p2 = &ikincideger; // p2 = ikincideger'in adresi
    *p1 = 10; // p1 ile gösterimdeki değer p1 = 10
    *p2 = *p1; // p2 ile gösterimdeki değer = p1 ile gösterimdeki
               //değer
    p1 = p2; // p1 = p2 (göstergeçin değeri kopyalandı)
    *p1 = 20; // p1 ile gösterimdeki = 20

    cout << "ilkdeger = " << ilkdeger << endl;
    cout << "ikincideger = " << ikincideger << endl;
    return 0;
}
```

0'dan 50 ye kadar olan Tek sayıların kumpen alan programı while döngüsü ile yazınız.

1 den 100 e kadar olan sayıların 3 ile bölümünden kalanı hesaplayan programı for döngüsü ile yazınız.

kişiyi cebindeki parayı sor. Para 20 tl den büyük olduğu sürece girdiği tüm paraları toplayan programı do while döngüsü ile yazınız.

```
#include <cstdlib>
#include <iostream>

using namespace std;

int main()
{

    cout<<"hosgeldiniz...9+8--"<<9+8<<"...fatih"<<endl;

    system("PAUSE");
    return EXIT_SUCCESS;
}
```

---

```
#include <cstdlib>
#include <iostream>

using namespace std;

#define bip '\a'

int main()
{
    string adiniz;
    int s1,s2,carp;
    char cins;
    cins = 'k';

    string cinsiyet = cins=='k' ? "kiz" : "erkek";

    cout<<"adinizi giriniz :";
    cin>>adiniz;
    cout<<"\n lutfen bir sayi girermisiniz :";
    cin>>s1;
    cout<<"\nbir sayi daha girermisiniz :";
    cin>>s2;

    carp = s1*s2+genislik;

    cout<<"\n sayin-"<<adiniz<<" sonucumuz-"<<carp<<endl;

    system("PAUSE");
    return EXIT_SUCCESS;
}
```

---



```
#include <cstdlib>
#include <iostream>

using namespace std;

int main()

{

    int kk,uk,alan;

    cout<<"kisa kenari giriniz :";
    cin>>kk;
    cout<<"\n uzun kenari giriniz :";
    cin>>uk;

    alan = kk*uk;

    cout<<"\n alan sonucumuz :"<<alan<<endl;


    system("PAUSE");
    return EXIT_SUCCESS;
}
```

---

```
#include <cstdlib>
#include <iostream>

#define PI 3.14
#define bip '\a'
using namespace std;

int main()

{

    int yariCap;
    float alan;
    cout<<"dairenin yaricapini giriniz :";
    cin>>yariCap;

    alan = PI * yariCap * yariCap;

    cout<<"\n alan sonucumuz :"<<alan<<bip<<bip<<bip<<bip<<endl;


    system("PAUSE");
    return EXIT_SUCCESS;
}
```

```

#include <cstdlib>
#include <iostream>

using namespace std;

int main()
{

    string adi;
    float kilo;
    float boy, vucutIndex;
    char karar;

    hsp:cout<<"Adinizi girer misiniz...";
    cin>>adi;
    cout<<"\nHosgeldiniz --"<<adi<<endl;

    cout<<"Kac kilosunuz :";
    cin>>kilo;
    cout<<"\nBoyunuz kac :";
    cin>>boy;

    vucutIndex = kilo/(boy*boy);

    cout<<"sayin "<<adi<<"----vucut kitle indeksiniz :"<<vucutIndex<<endl;

    if(vucutIndex <= 25)
    {
        cout<<"Kilonuz ve boyunuz ideal orantidadir..."<<endl;
    }

    else if(vucutIndex > 25 && vucutIndex < 30)
    {
        cout<<"Siskosunuz..."<<endl;
    }
    else
    {
        cout<<"\nAllah yardimcin olsun...\n"<<endl;
    }

    secim:cout<<"Tekrar hesaplamak ister misiniz ? (E/H)--";
    cin>>karar;

    if(karar == 'E' || karar == 'e')
    {
        goto hsp;
    }
    else if(karar == 'H' || karar == 'h')
    {
        goto cikis;
    }
    else

```

```

    {
        cout<<"\n yanlis giris yaptiniz H ya da E giriniz"<<endl;
        goto secim;
    }

    cikis:

    system("PAUSE");
    return EXIT_SUCCESS;
}

```

---

```

#include <cstdlib>
#include <iostream>
#include <string>

using namespace std;

int main()
{
    string ad,takim;
    int tns;
    cout<<" Adinizi girer misiniz ? ";
    getline(cin,ad);
    cout<<"\n Hangi takimi tutuyorsunuz ?";
    getline(cin,takim);

    cout<<"\n Tansiyonunuzun kucuk olani kac cikti ?";
    cin>>tns;

    cout<<"\n Hosgeldiniz--"<<ad<<"-- Tuttugunuz takim :"<<takim<<endl;

    if(tns <= 9)
    {
        cout<<"\n Tansiyon icin sor balik yiyin"<<endl;
    }
    else if (tns > 9 && tns <= 15)
    {
        cout<<"\n Tansiyon icin SIKINTI TINNE"<<endl;
    }
    else
    {
        cout<<"\n Tansiyon icin sarimsagi yut emraaah kosss"<<endl;
    }

    system("PAUSE");
    return EXIT_SUCCESS;
}

```

---

```

#include <cstdlib>
#include <iostream>

```

```
using namespace std;

int main()
{
    int fatih;
    cout<<"Bir deger giriniz--";
    cin>>fatih;

    while(fatih<20)
    {
        cout<<"girilen deger : "<<fatih<<endl;
        cout<<"girilen degerin 2 kati : "<<fatih*2<<endl;
        fatih++;
    }

    unsigned long ferhat;
    do {
        cout << "\n\nSayi giriniz (0 dan buyuk): ";
        cin >> ferhat;
        cout << "Girdiginiz sayi: " << ferhat << "\n";
    } while (ferhat != 0);

    for(int m=100; m>0; m--)
    {

        if(m > 50 && m < 65)
        {
            continue;
        }
        cout<<m<<endl;

    }

    system("PAUSE");
    return EXIT_SUCCESS;
}
```

---