

BMÜ-111 ALGORİTMA VE PROGRAMLAMA

AKIŞ KONTROLÜ

YRD. DOÇ. DR. İLHAN AYDIN

TEMEL AMAÇLAR

Bir dairenin yarıçapını kullanıcıdan alarak dairenin alanını hesaplamak istediğimizi düşünelim.

Eğer girilen değer negatif ise alan hesaplanamaz. Bu durumda ne yapmak gerekir.?

Aynı şekilde bir öğrencinin notunu kullanıcıdan istiyorsak ve kullanıcı 0-100 arası dışında bir değer girmişse ne yapmak gerekir?

BOOLEAN VERİ TÜRÜ VE OPERATÖRLER

Bir programda sık sık iki değeri karşılaştırmanız gerekebilir.

Örneğin «a b'den büyük müdür?» şeklinde bir karşılaştırma gerekebilir.

Java iki değeri karşılaştırmak için 6 karşılaştırma operatörü sunar.

Karşılaştırma sonucu Boolean bir değerdir: true (doğru) veya false (yanlış)

```
b = (1 > 2);
```

KARŞILAŞTIRMA OPERATÖRLERİ

Operator *isim*

< Küçük

<= Küçük veya eşit

> Büyük

>= Büyük veya eşit

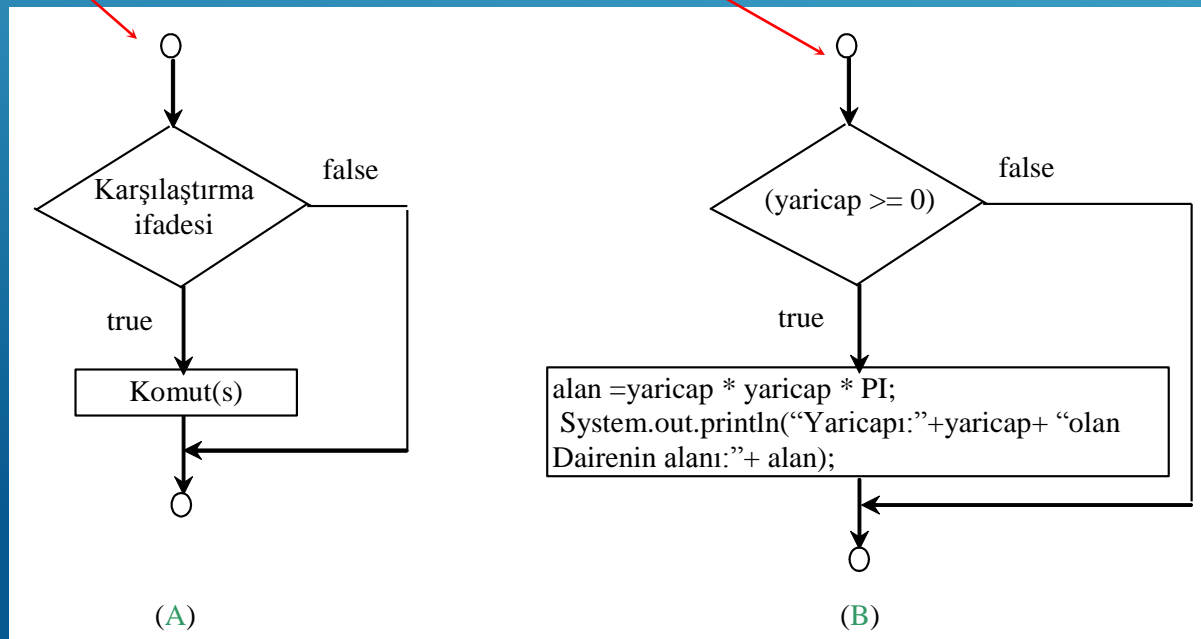
== Eşit

!= Eşit değil

IF KARAR YAPISI

```
if (yaricap >= 0) {  
    alan = yaricap * yaricap * PI;  
    System.out.println(«Yaricapi "  
        + yaricap + " olan dairenin alanı:" + area);  
}
```

```
if (boolean ifade) {  
    komut(s);  
}
```



NOT

```
if i > 0 {  
    System.out.println("I pozitiftir");  
}
```

(a) Yanlış

```
if (i > 0) {  
    System.out.println("i pozitiftir.");  
}
```

(b) Doğru

```
if (i > 0) {  
    System.out.println("i pozitiftir.");  
}
```

(a)

Eşdeğer

```
if (i > 0)  
    System.out.println("i pozitiftir.");
```

(b)

ÖRNEK

Kullanıcıdan tamsayı girmesini isteyen bir program yazınız. Eğer sayı 5'e bölünebiliyorsa «sayı 5'in katları» mesajını yazsın. Eğer sayı ikiye bölünüyorsa sayı çift mesajı yazsın

```
1 import java.util.Scanner;
2 public class Basitifyapisi {
3     public static void main(String[] args) {
4         Scanner klavye = new Scanner(System.in);
5         System.out.println(«Bir sayı girin: »);
6         int sayi = klavye. nextInt();
7         if (sayi%5==0)
8             System.out.println(sayi+ «sayısı»+ «5'in katı»);
9         if(sayi%2==0)
13            System.out.println(sayi+ « sayısı çift sayı");
14     }
```

ÖRNEK:

```
import java.util.Scanner;

public class Uygulama1{

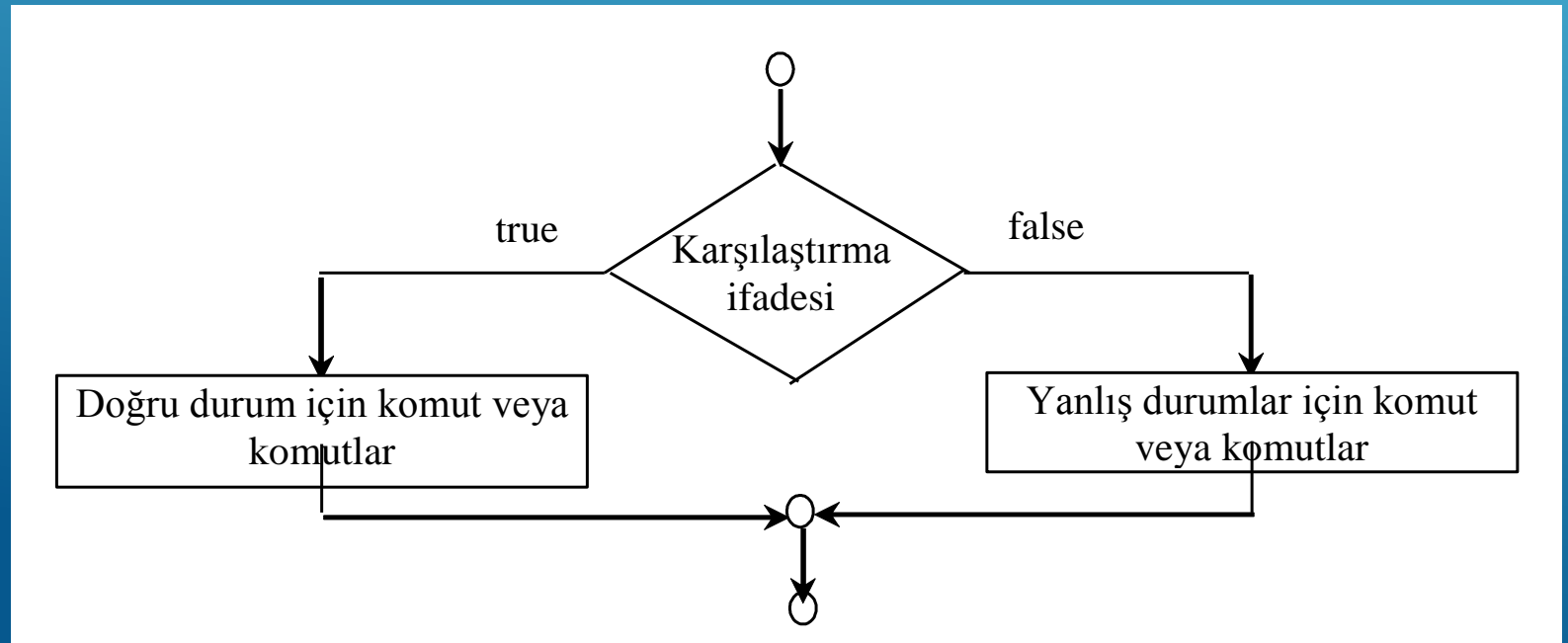
    public static void main(String[] args) {
        Scanner klavye=new Scanner(System.in);
        String ileti="Bilgisayarı kullanmak işleri kolaylaştırır";
        int uzunluk=ileti.length();
        if(uzunluk<50)
            System.out.println("kısa bir cümle");
        }
    }
```


ÖRNEK: Girilen üç karakteri küçükten büyüğe sıralayan program

```
2  import java.util.Scanner;
3  public class Uygulama1{
4      public static void main(String[] args) {
5          Scanner klavye=new Scanner(System.in);
6          char ara;
7          System.out.println("üç karakter giriniz");
8          char a=klavye.next().charAt(0);
9          char b=klavye.next().charAt(0);
10         char c=klavye.next().charAt(0);
11         if(a>b) {
12             ara=a;      a=b;      b=ara;
13         }
14         if(a>c) {
15             ara=a;      a=c;      c=ara;
16         }
17         if(b>c) {
18             ara=b;      b=c;      c=ara;
19         }
20         System.out.println("sıralanmış karakterler:"+a+ " "+b+" "+c);
21     }
22 }
```

İKİ YOLLU İF YAPISI (IF-ELSE YAPISI)

```
if (karşılaştırma ifadesi) {  
    Komut(s)-şart doğru ise  
}  
else {  
    komut(s)-şart yanlış ise  
}
```



İF...ELSE ÖRNEK

```
if (yaricap >= 0) {  
    alan = yaricap * yaricap * 3.14159;  
  
    System.out.println(«Yarıcapı:“+ yaricap+ «olan dairenin  
    alanı:»+alan);  
}  
else {  
    System.out.println("Negatif giriş");  
}
```

ÖRNEK:

```
import java.util.Scanner;

public class Uygulama1{

    public static void main(String[] args) {
        Scanner klavye=new Scanner(System.in);
        System.out.println("üçgenin kenarlarını giriniz:");
        double x=klavye.nextDouble();
        double y=klavye.nextDouble();
        double z=klavye.nextDouble();
        if((x+y)>z)
            System.out.println("kenarlar bir üçgen oluşturur");
        else
            System.out.println("kenarlar üçgen oluşturamaz");
    }
}
```

ÖRNEK: Kenarları girilen bir üçgenin ne tür bir üçgen olduğunu bulan java programı

```
3  import java.util.Scanner;
4  public class Uygulama1{
5      public static void main(String[] args) {
6          Scanner klavye=new Scanner(System.in);
7          int a,b,c;
8          System.out.println("Üç kenar giriniz");
9          a=klavye.nextInt();
10         b=klavye.nextInt();
11         c=klavye.nextInt();
12         if(a==b && b==c) System.out.println("Eşkenar üçgen");
13         else if (a==b || b==c || a==c)
14             System.out.println("ikiz kenar üçgen");
15         else System.out.println("çeşit kenar üçgen");
16     }
17 }
```

İF ŞARTININ ÇOKLU ALTERNATİFLERİ

```
if (puan >= 90.0)
    sonuc = 'A';
else
    if (puan >= 80.0)
        sonuc = 'B';
    else
        if (puan >= 70.0)
            sonuc = 'C';
        else
            if (puan >= 60.0)
                sonuc = 'D';
            else
                sonuc = 'F';
```

Eşdeğer

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```

IF-ELSE YAPISI İÇİN ÖRNEK

puan değişkeni 70.0 olsun

Şart yanlış

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```

IF-ELSE YAPISI İÇİN ÖRNEK

puan değişkeni 70.0 olsun

Şart yanlış

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```


IF-ELSE YAPISI İÇİN ÖRNEK

puan değişkeni 70.0 olsun

Şart doğru

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```

IF-ELSE YAPISI İÇİN ÖRNEK

puan değişkeni 70.0 olsun

sonuc C

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```

IF-ELSE YAPISI İÇİN ÖRNEK

puan değişkeni 70.0 olsun

```
if (puan >= 90.0)
    sonuc = 'A';
else if (puan >= 80.0)
    sonuc = 'B';
else if (puan >= 70.0)
    sonuc = 'C';
else if (puan >= 60.0)
    sonuc = 'D';
else
    sonuc = 'F';
```

if deyiminden çıkış

NOT

else tümcesi aynı blokta ise en son if tümcesi ile eşleşir.

```
int i = 1;
int j = 2;
int k = 3;

if (i > j)
    if (i > k)
        System.out.println("A");
else
    System.out.println("B");
```

(a)

Eş değer

```
int i = 1;
int j = 2;
int k = 3;

if (i > j)
    if (i > k)
        System.out.println("A");
else
    System.out.println("B");
```

(b)

NOT

Önceki deyimden ekrana herhangi bir şey yazılmaz. Eğer else yapısının birinci if yapısının yanlış olması durumunda çalışmasını istiyorsak { } parantezlerini kullanmalıyız.

```
int i = 1;  
int j = 2;  
int k = 3;  
if (i > j) {  
    if (i > k)  
        System.out.println("A");  
}  
else  
    System.out.println("B");
```

Bu ifade ekrana B yazar.

EN ÇOK YAPILAN HATALAR

if deyiminden sonra noktalı virgül koymak en çok yapılan hatalardandır.

Yanlış

```
if (yaricap >= 0);  
{  
    alan = yaricap*yaricap*PI;  
    System.out.println(  
        «Yaricapı: " +yaricap+ " olan dairenin alanı: " + alan);  
}
```

Bu hatayı bulmak zordur, çünkü çalışma zamanı veya derleme hatası vermez, bu mantıksal bir hatadır.

ATAMA İLE KARŞILAŞTIRMA

```
if (sayi % 2 == 0)
    cift = true;
else
    cift = false;
```

(a)

Eşdeğer

```
boolean cift
    = sayi % 2 == 0;
```

(b)

Dikkat

```
if (cift == true)
    System.out.println(
        "sayı çift.");
```

(a)

Eşdeğer

```
if (cift)
    System.out.println(
        "sayı çift.");
```

(b)

MANTIKSAL OPERATÖRLER

<i>Operatör</i>	<i>İsim</i>
-----------------	-------------

!	değil
---	-------

& &	ve
-----	----

	veya
--	------

^	özel veya
---	-----------

DEĞİL (!) OPERATÖRÜ İÇİN TABLO

p	!p	Örnek (yas = 24, cinsiyet = 'E' olduğunu varsayalım)
true	false	!(yas > 18) false dir, Çünkü (age > 18) true dur.
false	true	!(cinsiyet!= 'E') true dir, Çünkü (cinsiyet != 'E') false dir..

p	!p	Örnek
true	false	!(1 > 2) true degeri alır, çünkü (1 > 2) yanlıştır.
false	true	

&& OPERATÖRÜ İÇİN DOĞRULUK TABLOSU

p1	p2	p1 && p2	Örnek (yas = 24, cinsiyet = 'K')
false	false	false	<u>(yas > 18) && (cinsiyet == 'K')</u> is true, çünkü <u>(yas > 18)</u> ve <u>(cinsiyet == 'K')</u> şartlarının her ikisi de true değeri alır.
false	true	false	
true	false	false	
true	true	true	<u>(yas > 18) && (cinsiyet != 'K')</u> false değeri alır, çünkü <u>(cinsiyet != 'K')</u> false değeri alır.

p1	p2	p1 && p2	Örnek
false	false	false	(3 > 2) && (5 >= 5) true değeri alır, çünkü (3 > 2) ve (5 >= 5) her ikisi de true değeri alır.
false	true	false	
true	false	false	
true	true	true	(3 > 2) && (5 > 5) ise false değeri alır, çünkü (5 > 5) false değeri alır.

|| OPERATÖRÜ İÇİN DOĞRULUK TABLOSU

p1	p2	p1 p2	Örnek(yas= 24, cinsiyet= 'K')
false	false	false	<u>(yas > 34) (cinsiyet == 'K')</u> true degeri alır, çünkü <u>(cinsiyet == 'K')</u> true degeri alır.
false	true	true	
true	false	true	<u>(yas > 34) (cinsiyet == 'E')</u> false degeri alır, çünkü <u>(yas > 34)</u> ve <u>(cinsiyet == 'E')</u> her ikisi de false degeri alır.
true	true	true	

p1	p2	p1 p2	Örnek
false	false	false	$(2 > 3) (5 > 5)$ false degeri alır, çünkü $(2 > 3)$ ve $(5 > 5)$ in her ikisi de yanlış degeri alır.
false	true	true	
true	false	true	$(3 > 2) (5 > 5)$ true degeri alır, çünkü $(3 > 2)$ true degeri alır.
true	true	true	

^ OPERATÖRÜ İÇİN DOĞRULUK TABLOSU

p1	p2	p1 ^ p2	Örnek (yas = 24, cinsiyet = 'K')
false	false	false	<u>(yas > 34) ^ (cinsiyet == 'K')</u> true deger alır, cunku <u>(yas > 34)</u> false degeri alırken <u>(cinsiyet == 'K')</u> true deger alır.
false	true	true	
true	false	true	
true	true	false	<u>(yas > 34) (cinsiyet == 'E')</u> false deger alır, cunku <u>(yas > 34)</u> ve <u>(cinsiyet == 'E')</u> her ikisi de

ÖRNEK

```
import java.util.Scanner;
public class Uygulama1{
    public static void main(String[] args) {
        Scanner klavye=new Scanner(System.in);
        System.out.println("ortalamanızı giriniz:");
        double ort=klavye.nextDouble();
        if(ort>4)
            System.out.println("hatali ortalama");
        else if(ort<=2.0)
            System.out.println("sinifta kaldınız");
        else if(ort>=3.0)
            System.out.println("Onur listesine girdiniz");
        else if((ort>2)&&(ort<3.0))
            System.out.println("sinifi geçtiniz");
    }
}
```

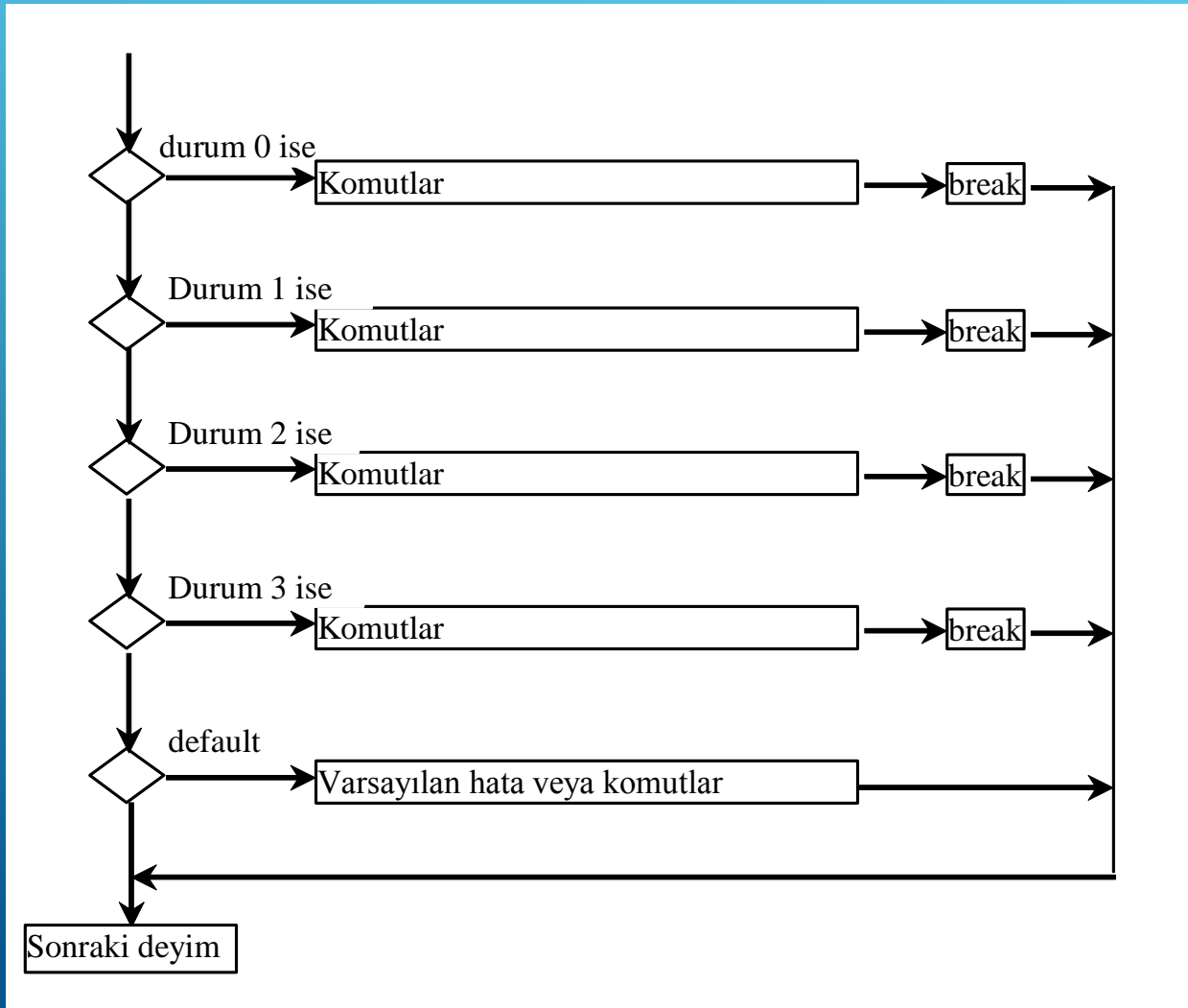
ÖRNEK

```
import java.util.Scanner;
public class Uygulama1{
    public static void main(String[] args) {
        Scanner klavye=new Scanner(System.in);
        System.out.println("Bir sayı giriniz");
        int sayi=klavye.nextInt();
        if(sayi%5==0 && sayi%6==0)
            System.out.println("sayı 5 ve 6 ya bölünür");
        else if(sayi%5==0)
            System.out.println("Sayı sadece 5 e bölünür");
        else if(sayi%6==0)
            System.out.println("sayı sadece 6 ya bölünür");
        else
            System.out.println("sayı ikisine de bölünmez");
    }
}
```

SWITCH DEYİMİ

```
switch (denetim ifadesi) {  
    case etiket1: komutlar;  
        break;  
    case etiket2: komutlar;  
        break;  
    case etiket3: komutlar;  
        break;  
    case etiket4: komutlar;  
        break;  
    default: System.out.println(«Hatalar: Geçersiz etiket»);  
        System.exit(0);  
}
```

SWITCH DEYİMİ AKIŞ ŞEMASI



SWITCH STATEMENT RULES

Switch deyimi parantez içinde daima char, byte, short, veya int veri türlerinden birini almalı.

deger1, ..., degerN switch-deyimi ile girilen değer ile aynı türden olmalıdır. Switch-deyimi ile verilen ifade case ile belirlenen değerle eşleşiyorsa ilgili case'ten sonraki komut çalışır.

Not: deger1, ..., ve degerN sabit ifadelerdir. Değişken olmamaları gerekir. Örneğin $1+x$ şeklinde bir ifade yanlıştır.

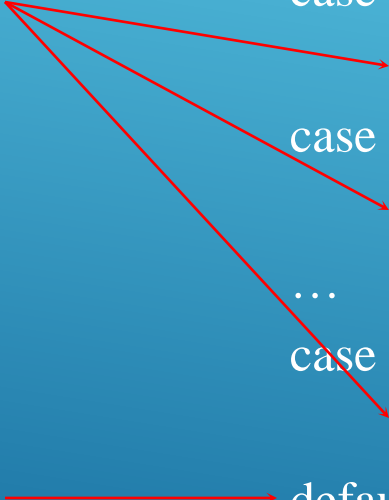
```
switch (switch-deyimi) {  
    case deger1: statement(s)1;  
        break;  
    case deger2: statement(s)2;  
        break;  
    ...  
    case degerN: statement(s)N;  
        break;  
    default: statement(s)-for-default;  
}
```

SWITCH DEYİMİ KURALLARI

break anahtarı isteğe bağlıdır. Fakat switch deyiminin geri kalanını sonlandırmak için kullanılması gerekir. Eğer break deyimi verilmezse sonraki durum ifadesi yürütülecektir.

default durumu isteğe bağlıdır. Switch deyimindeki ifade hiçbir durum ile eşleşmez ise uygulanacak hareketleri belirler.

```
switch (switch-expression) {  
    case value1: statement(s)1;  
        break;  
    case value2: statement(s)2;  
        break;  
    ...  
    case valueN: statement(s)N;  
        break;  
    default: statement(s)-for-default;  
}
```

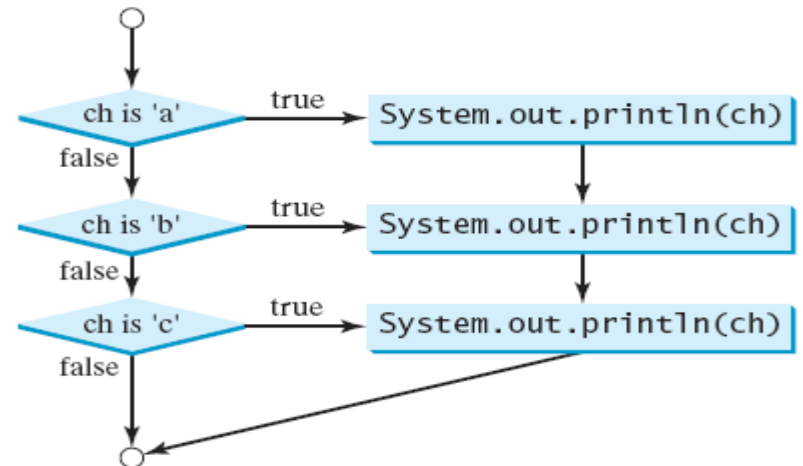


case deyimleri ardışık bir düzende yürütülür, fakat case'lerin sırası (default da dahil) önemli değildir. Fakat case'lerin mantıksal sırasını izlemek ve default ile switch'i sonlandırmak iyi bir programlama stilidir.

ÖRNEK SWITCH UYGULAMASI

ch değişkeninin değerinin 'a' olduğunu varsayalım:

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```



Örnek Switch uygulaması

ch 'a' değerine eşit:

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```

Örnek Switch uygulaması

Bu satırı işlet

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```

Örnek Switch uygulaması

Bu satırı işlet

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```

ÖRNEK SWITCH UYGULAMASI

Bu satırı işlet

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```

ÖRNEK SWITCH UYGULAMASI

Sonraki deyimi çalıştır

```
switch (ch) {  
    case 'a': System.out.println(ch);  
    case 'b': System.out.println(ch);  
    case 'c': System.out.println(ch);  
}
```

Sonraki deyim;

ÖRNEK SWITCH UYGULAMASI

ch değişkeninin değerinin 'a' olduğunu varsayalım:

```
switch (ch) {  
    case 'a': System.out.println(ch) ;  
                break;  
    case 'b': System.out.println(ch) ;  
                break;  
    case 'c': System.out.println(ch) ;  
}
```

ÖRNEK SWITCH UYGULAMASI

ch değişkeni 'a' ya eşit

```
switch (ch) {  
    case 'a': System.out.println(ch);  
               break;  
    case 'b': System.out.println(ch);  
               break;  
    case 'c': System.out.println(ch);  
}
```

ÖRNEK SWITCH UYGULAMASI

Bu komutu çalıştır

```
switch (ch)
{
    case 'a': System.out.println(ch);
               break;
    case 'b': System.out.println(ch);
               break;
    case 'c': System.out.println(ch);
}
}
```

ÖRNEK SWITCH UYGULAMASI

Bu komutu çalıştır

```
switch (ch) {  
    case 'a': System.out.println(ch);  
               break;  
    case 'b': System.out.println(ch);  
               break;  
    case 'c': System.out.println(ch);  
}
```

ÖRNEK SWITCH UYGULAMASI

Sonraki komutu çalıştır

```
switch (ch) {  
    case 'a': System.out.println(ch) ;  
               break;  
    case 'b': System.out.println(ch) ;  
               break;  
    case 'c': System.out.println(ch) ;  
}
```

Sonraki komut;

ÖRNEK:

```
2  import java.util.Scanner;
3  public class Uygulama1{
4      public static void main(String[] args) {
5          Scanner klavye=new Scanner(System.in);
6          int a,b;
7          String c;
8          System.out.println("iki tamsayı giriniz");
9          a=klavye.nextInt();
10         b=klavye.nextInt();
11         System.out.println("bir operatör giriniz:(+, -, *, /)");
12         c=klavye.next();
13         switch(c) {
14             case "+": System.out.println("Toplam:"+(a+b)); break;
15             case "-": System.out.println("Fark:"+(a-b)); break;
16             case "*": System.out.println("Çarpım:"+(a*b)); break;
17             case "/": System.out.println("Bölüm:"+((double)a/b)); break;
18             default: System.out.println("Yanlış operatör girdiniz");
19         }
    }
```

ŞART OPERATÖRÜ

```
if (x > 0)
```

```
    y = 1
```

```
else
```

```
    y = -1;
```

Şartı ile

$y = (x > 0) ? 1 : -1;$ şartı aynı işi görür.

$(\text{mantıksal-ifade}) ? \text{deyim1} : \text{deyim2}$

ŞART OPERATÖRÜ

```
if (sayi % 2 == 0)
```

```
    System.out.println(sayi + “ çifttir”);
```

```
else
```

```
    System.out.println(sayi + “tektir”);
```

```
System.out.println(
```

```
    (sayi % 2 == 0)? sayi + “ çifttir.” :
```

```
    sayi + “ tektir”);
```


FORMATLI ÇIKIŞ

Formatlı çıkış için printf ifadesi kullanılır. Burada amaç çıkışta gösterilecek değişkenin türünü formatlı ifade ile belirlemektir.

```
System.out.printf(format, degsikenler);
```

Burada format formatlı ifade ve alt string içeren bir string ifadedir.

Format tanımlayıcı bir değişkenin nasıl gösterileceğinin belirler.

Bir nesne numerik, karakter, boolean, veya string olabilir.

Her format tanımlayıcısı % ifadesi ile başlar

SIKLIKLA KULLANILAN TANIMLAYICILAR

Tanımlayıcı çıkış

<u>%b</u>	boolean bir değer
<u>%c</u>	karakter
<u>%d</u>	tam sayı
<u>%f</u>	noktalı sayı
<u>%e</u>	standart bilimsel gösterimde bir sayı
<u>%s</u>	Bir sayı

Example

true veya false

'a'

200

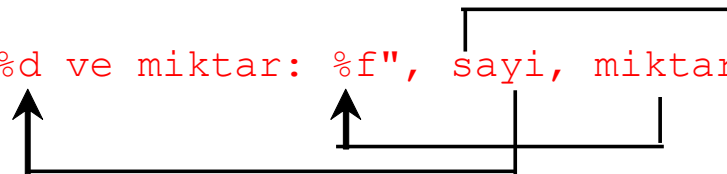
45.460000

4.556000e+01

"Java Programlama"

```
int sayi = 5;  
double deger = 45.56;  
System.out.printf("sayı: %d ve miktar: %f", sayi, miktar);
```

nesneler



Çıktı

sayı: 5 ve miktar: 45.560000