

Percabangan dan Perulangan

Yuliana Setiowati
Politeknik Elektronika Negeri Surabaya
2007






Topik

- Percabangan
- switch
- Perulangan
- Special Loop Control





Flowchart

- Bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.
- Merupakan cara penyajian dari suatu algoritma.

Simbol-Simbol Flowchart

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/ pemberian harga awal
	PROSES	Proses perhitungan/ proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi

Symbol-Symbol Flowchart

SIMBOL	NAMA	FUNGSI
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/ proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

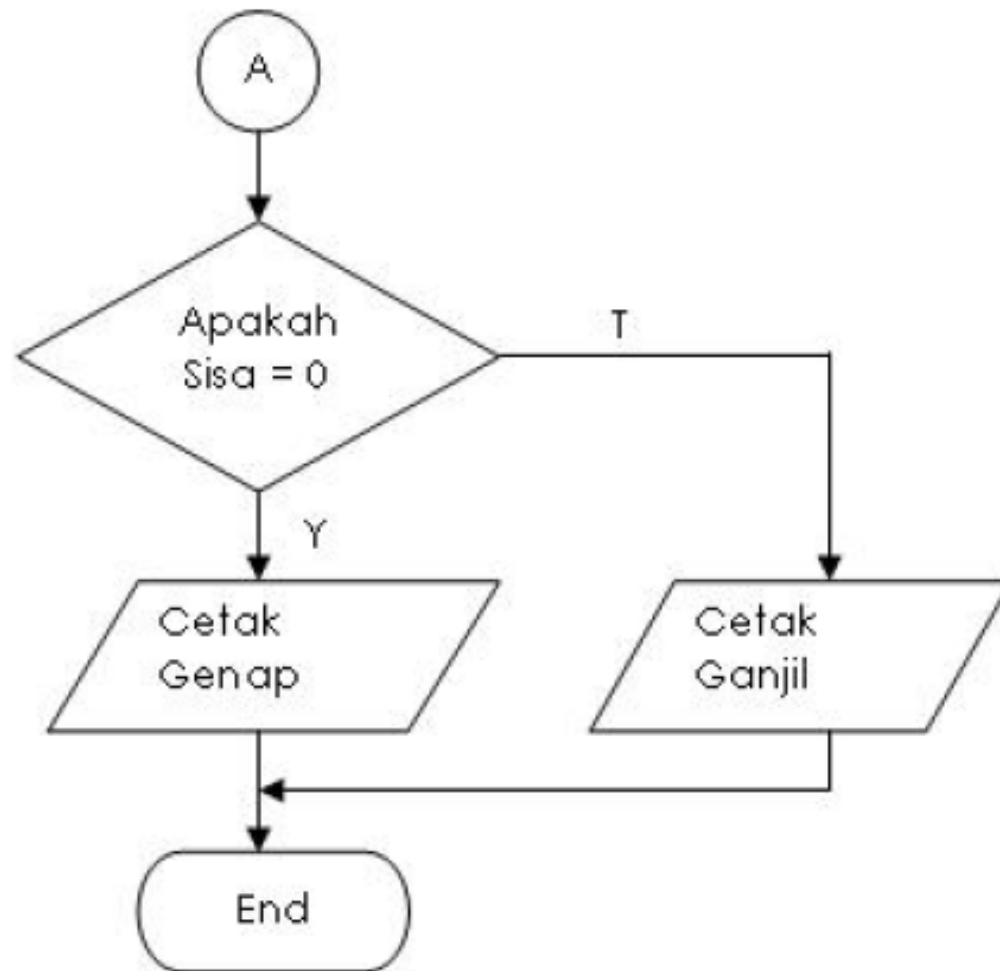
Pembuatan Flowchart

- Tidak ada kaidah yang baku.
- Flowchart adalah gambaran hasil analisa suatu masalah
- Flowchart dapat bervariasi antara satu pemrogram dengan pemrogram lainnya.
- Secara garis besar ada 3 bagian utama:
 - Input
 - Proses
 - Output

Pembuatan Flowchart

- Hindari pengulangan proses yang tidak perlu dan logika yang berbelit sehingga jalannya proses menjadi singkat.
- Jalannya proses digambarkan dari atas ke bawah dan diberikan tanda panah untuk memperjelas.
- Sebuah flowchart diawali dari satu titik START dan diakhiri dengan END.

Pembuatan Flowchart

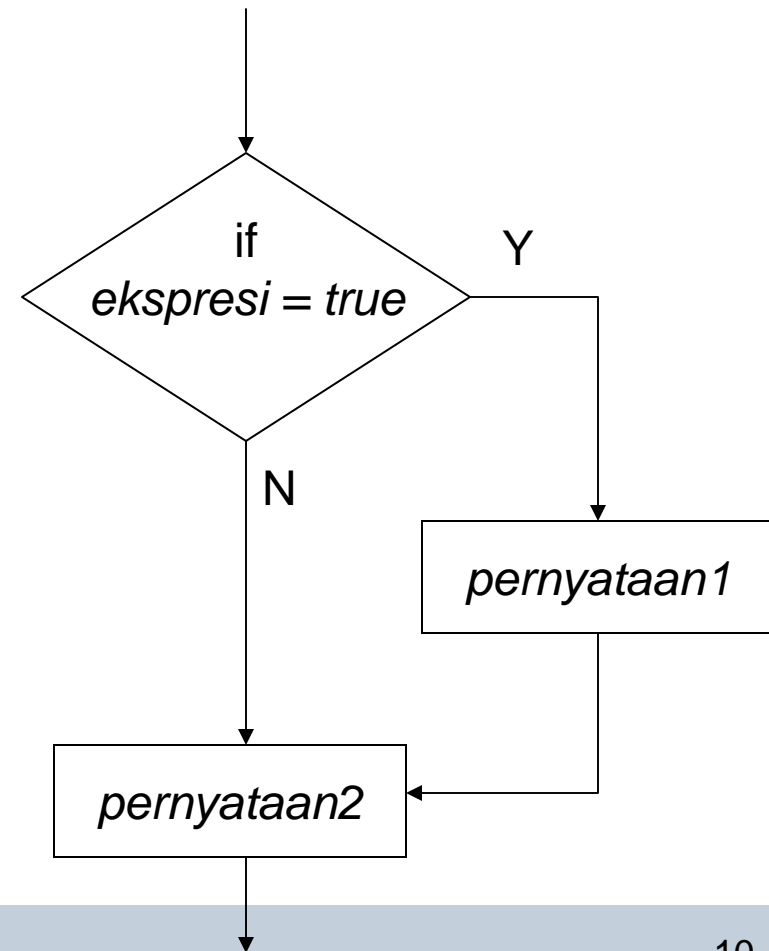


Percabangan

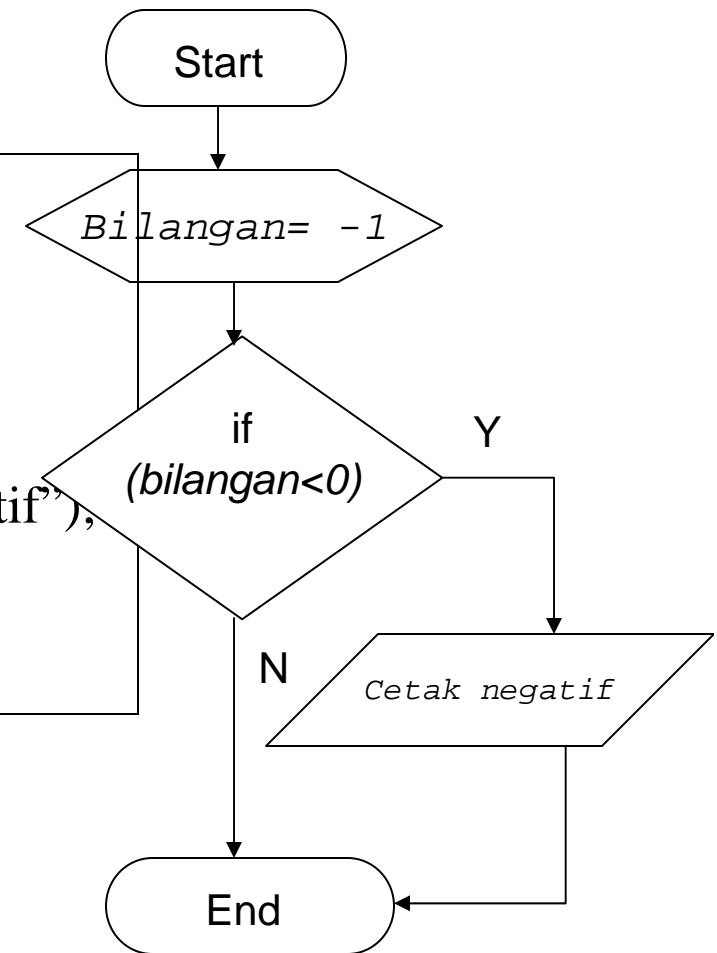
- if
- if-else
- else-if
- switch

if

```
if (ekspresi boolean) {  
    pernyataan1;  
}  
pernyataan2;
```



```
public class If {  
    public static void main(String args[] ) {  
        int bilangan=-1;  
        if (bilangan<0)  
            System.out.println("Bilangan adalah negatif"),  
    }  
}
```



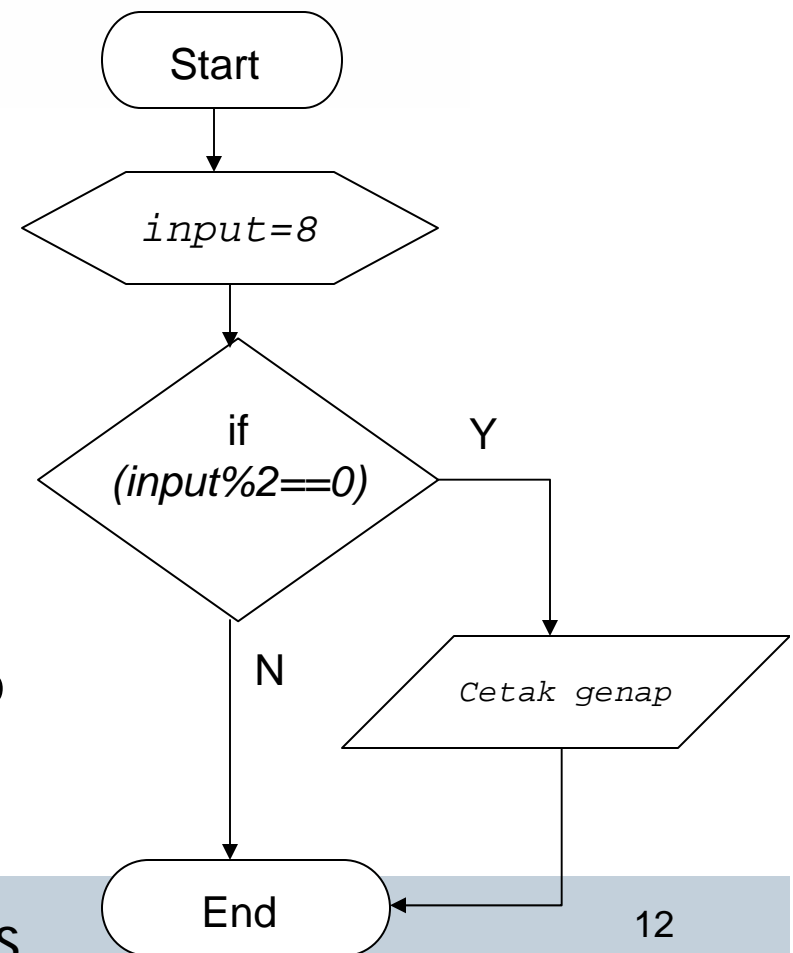
- **Output**
Bilangan adalah negatif



```

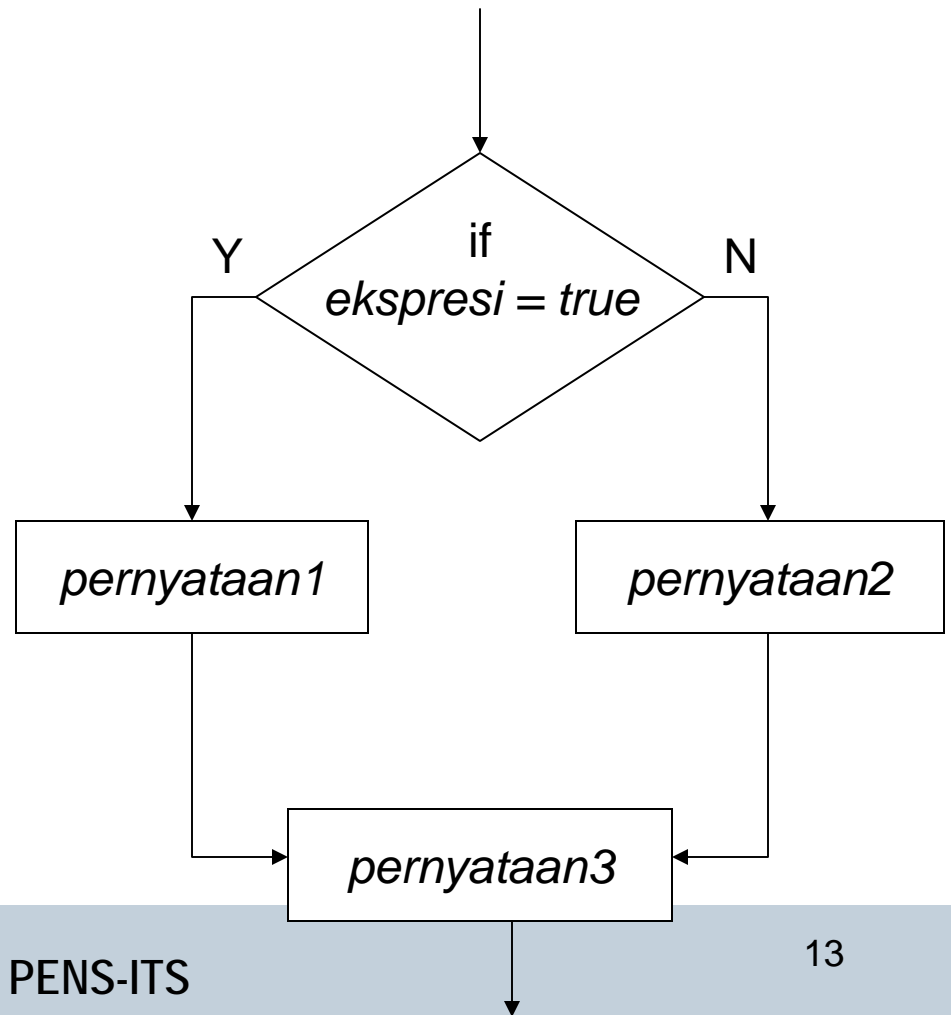
1 public class if2{
2     public static void main(String args[]){
3         int input = 8 ;
4         if (input %2 == 0)
5             System.out.println(input + " adalah Bilangan Genap");
6     }
7 }
    
```

- **Output**
8 adalah Bilangan Genap



if-else

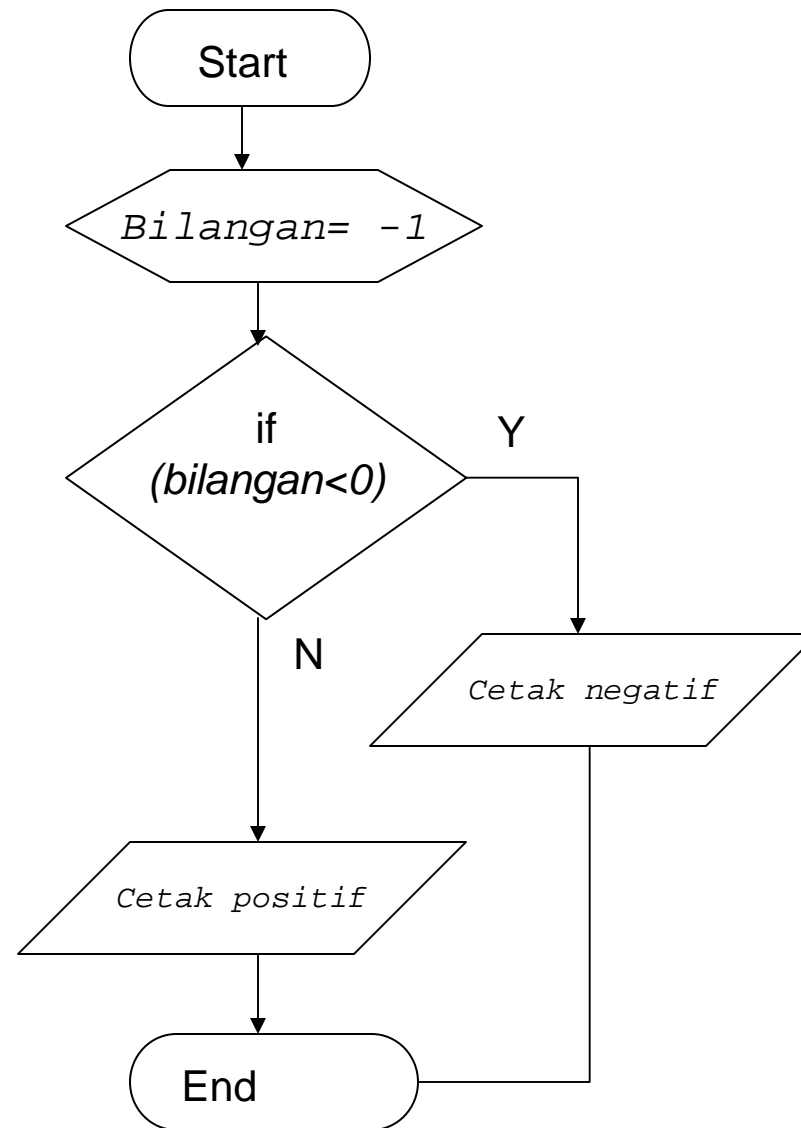
```
if (ekspresi boolean) {  
    pernyataan1;  
} else {  
    pernyataan2;  
}  
pernyataan3;
```



Contoh Program

```
public class IfElse {  
    public static void main(String args[]) {  
        int bilangan=-1;  
        if (bilangan<0)  
            System.out.println("Bilangan adalah negatif");  
        else  
            System.out.println("Bilangan adalah positif");  
    }  
}
```

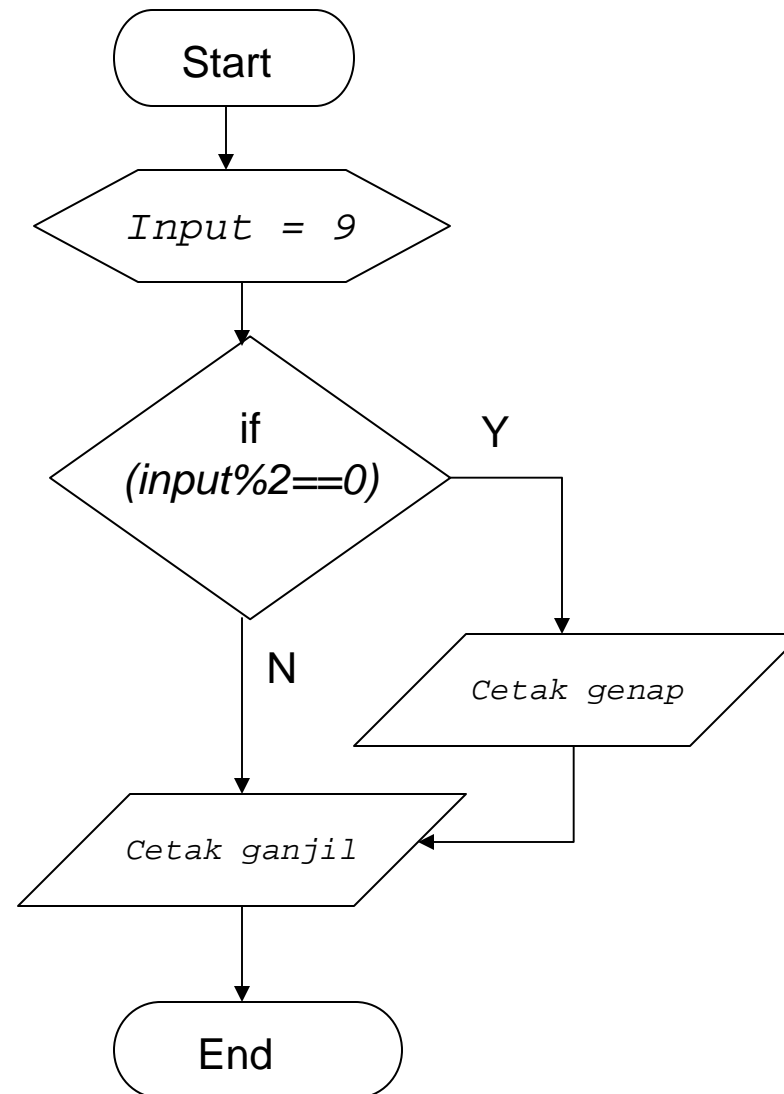
- **Output**
Bilangan adalah negatif



Contoh Program

```
1 public class if2{
2     public static void main(String args[]){
3         int input = 9 ;
4         if (input %2 == 0)
5             System.out.println(input + " adalah Bilangan Genap");
6         else
7             System.out.println(input + " adalah Bilangan Ganjil");
8     }
9 }
```

- **Output :**
9 adalah Bilangan Ganjil



Contoh Program

- User memasukkan inputan dengan menggunakan keyboard

- `import java.util.Scanner;`
- `public class if32 {`
- `public static void main(String args[]){`
- `Scanner scan = new Scanner(System.in);`
- `System.out.print("Masukkan Bilangan = ");`
- `int input = scan.nextInt();`
- `if (input %2 == 0)`
- `System.out.println(input + " adalah Bilangan Genap");`
- `else`
- `System.out.println(input + " adalah Bilangan Ganjil");`
- `}`
- `}`

Output :

```
Masukkan Bilangan = 3
3 adalah Bilangan
Ganjil
```

Output ?

- `public class if2{`
- `public static void main(String args[]){`
- `int input = 9 ;`
- `if (input %2 == 0)`
- `System.out.println(input + " adalah Bilangan Genap");`
- `else`
- `System.out.println(input + " adalah Bilangan Ganjil");`
- `System.out.println("Selalu dijalankan");`
- `}`
- `}`

If - else

- *if (ekspresi boolean)* bukan numerik

- Contoh :

```
if (x) // x integer
```

```
gunakan if (x!=0)
```



Contoh program

- `import java.util.Scanner;`
- `public class ifTebakAngka {`
- `public static void main(String args[]){`
- `Scanner scan = new Scanner(System.in);`
- `System.out.print("Masukkan Bilangan = ");`
- `int angka = scan.nextInt();`
- `if (angka != 15)`
- `System.out.println("Tebakan Anda Salah");`
- `else`
- `System.out.println("Tebakan Anda Benar");`
- `}`
- `}`

Output :

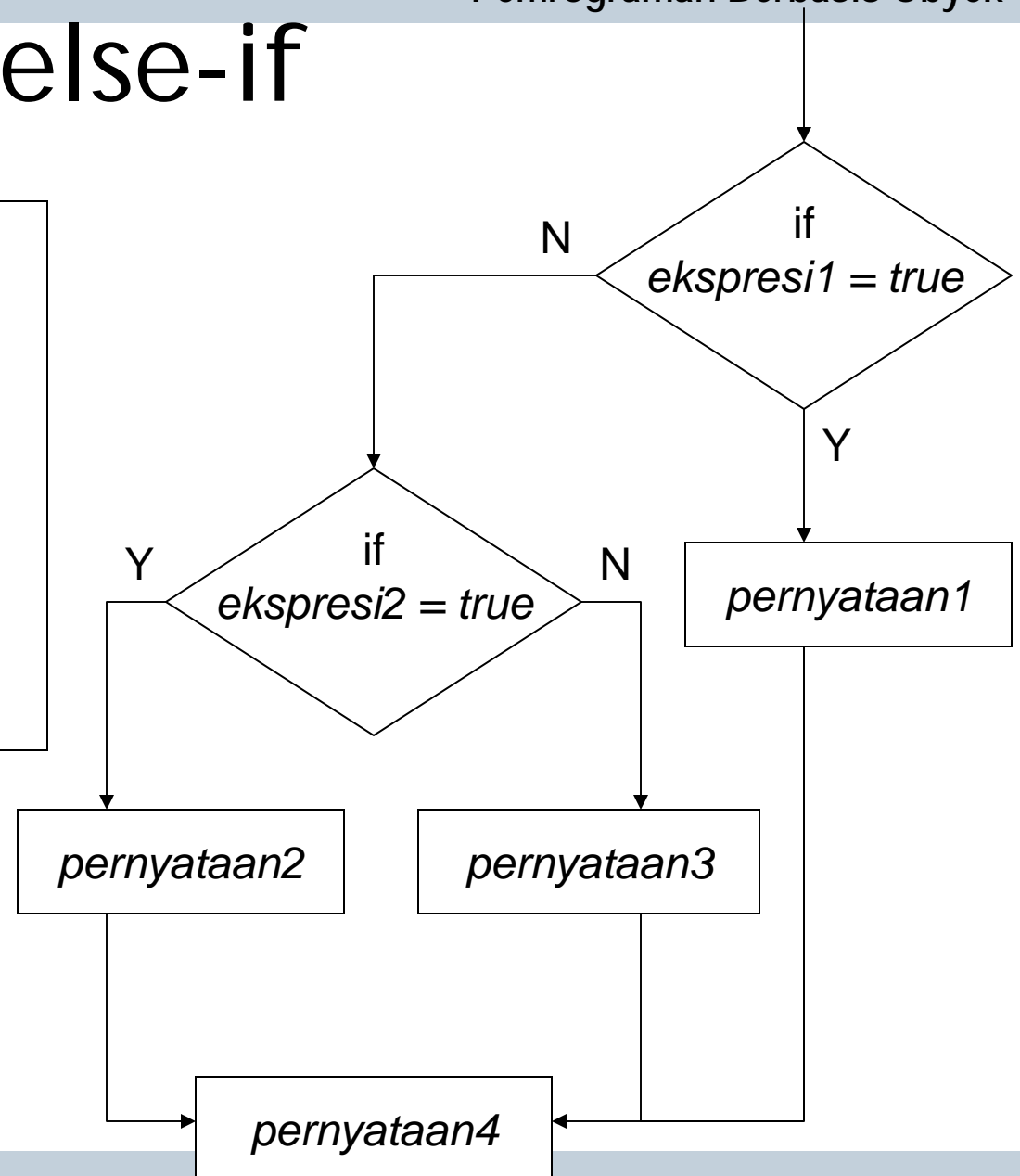
```
Masukkan Bilangan = 1  
Tebakan Anda Salah
```

Output :

```
Masukkan Bilangan = 15  
Tebakan Anda Benar
```

else-if

```
if (ekspresi1 boolean) {  
    pernyataan1;  
} else if (ekspresi2 boolean) {  
    pernyataan2;  
} else {  
    pernyataan3;  
}  
pernyataan4;
```



Contoh Program

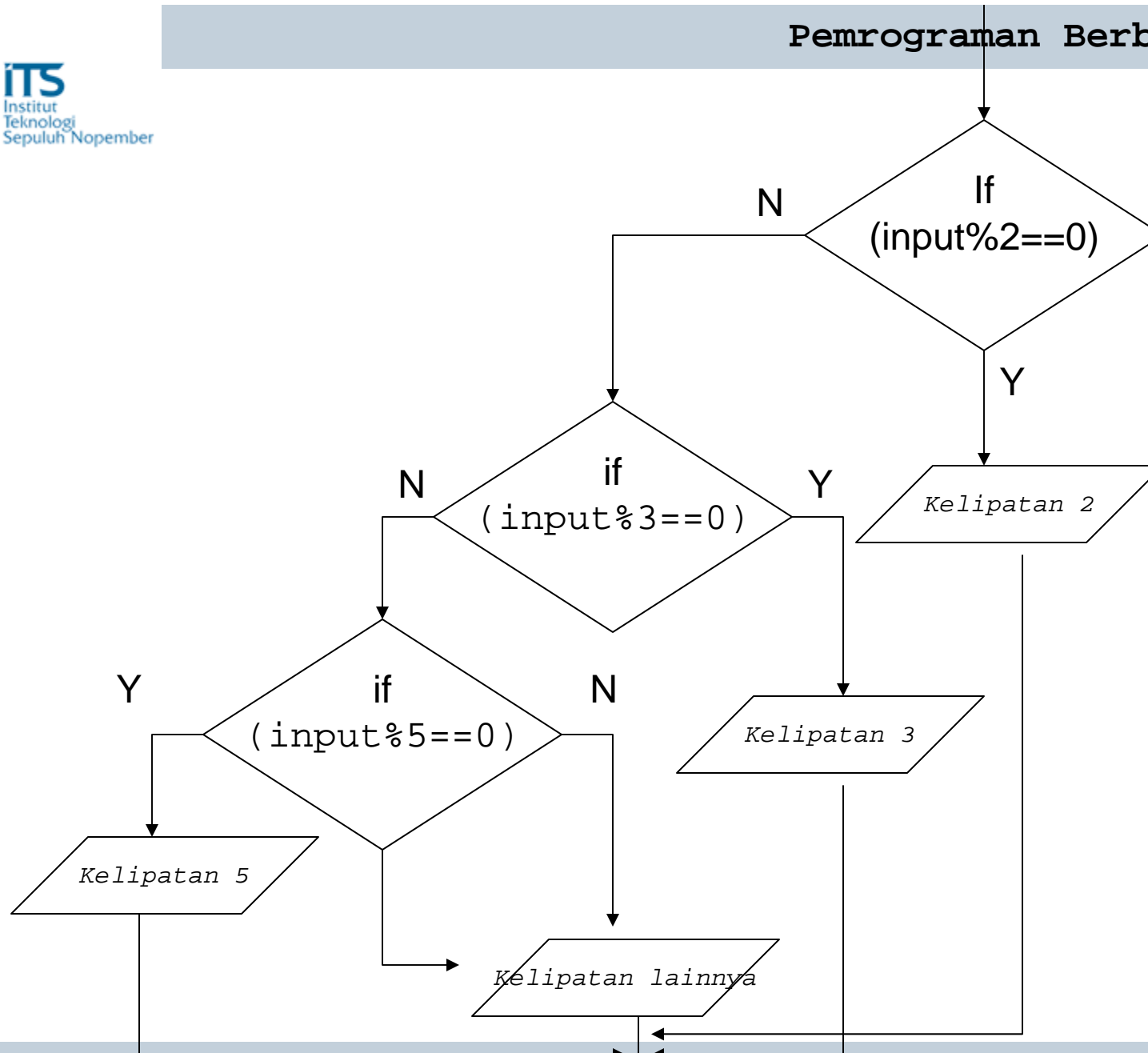
```
• import java.util.Scanner;
• public class if42 {
•     public static void main(String args[]){
•         Scanner scan = new Scanner(System.in);
•         System.out.print("Masukkan Bilangan = ");
•         int input = scan.nextInt() ;

•         if (input %2 == 0)
•             System.out.println(input + " adalah kelipatan 2");
•         else if (input %3 == 0)
•             System.out.println(input + " adalah kelipatan 3");
•         else if (input %5 == 0)
•             System.out.println(input + " adalah kelipatan 5");
•         else
•             System.out.println(input + " adalah kelipatan lainnya");
•     }
• }
```

```
Masukkan Bilangan =8
8 adalah kelipatan 2
```

```
Masukkan Bilangan =9
9 adalah kelipatan 3
```

```
Masukkan Bilangan =10
10 adalah kelipatan 5
```

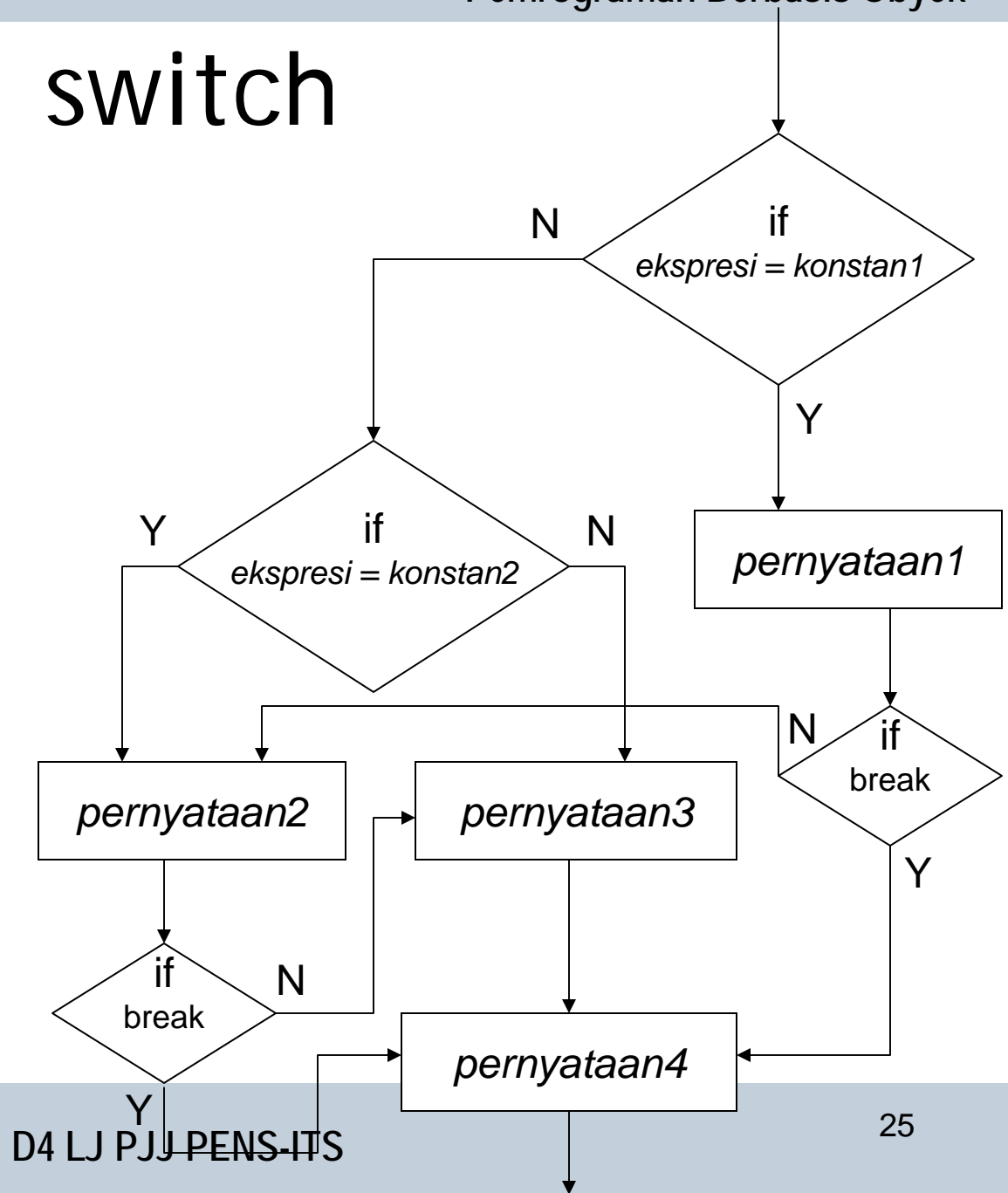


switch

```

switch (ekspresi) {
  case konstan1 :
    pernyataan1;
    break;
  case konstan2:
    pernyataan2;
    break;
  default :
    pernyataan3;
}
pernyataan4;

```



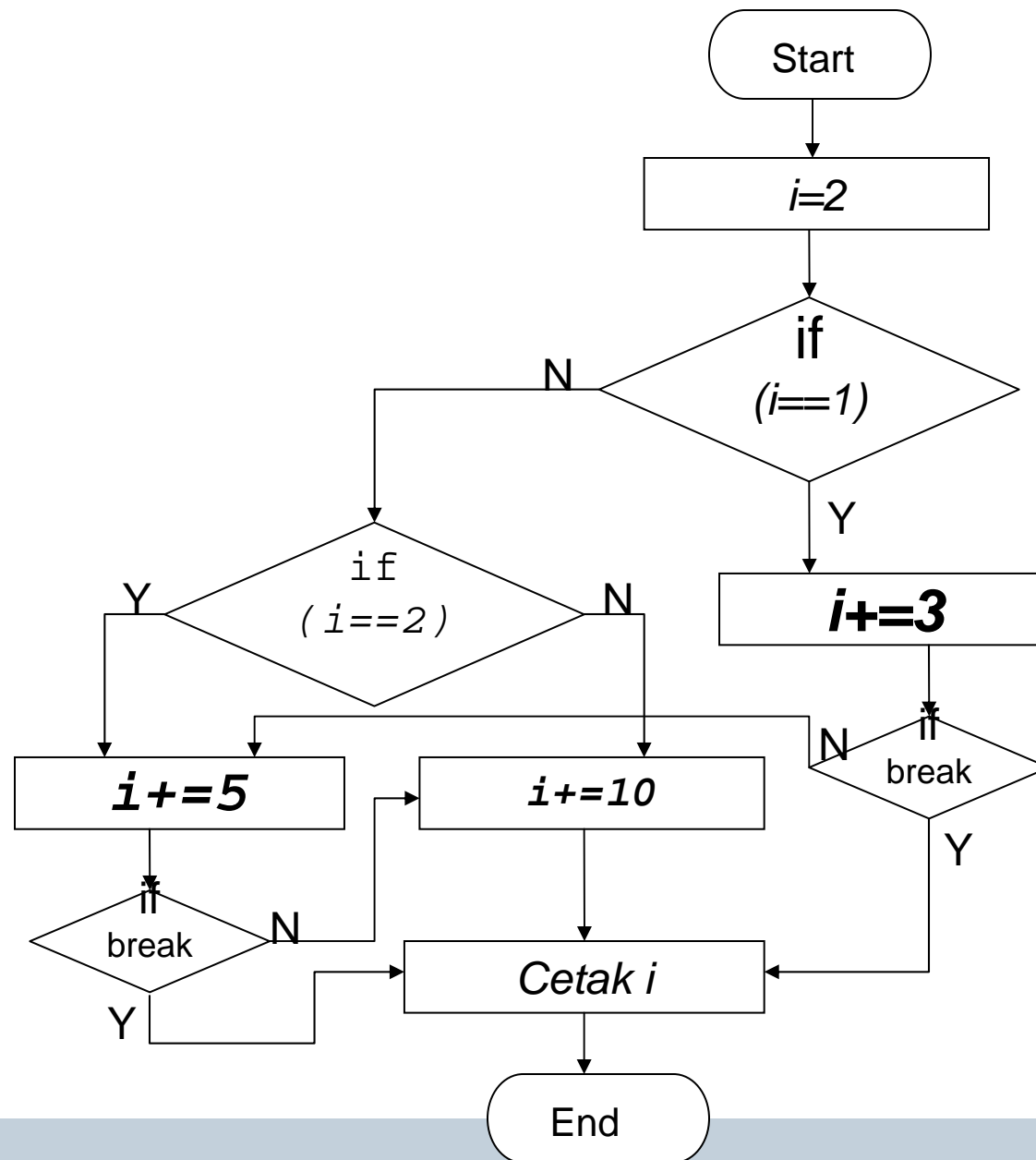
switch

- Variabel x harus bertipe byte, short, char, atau int.
- Floating point, long, atau class references (termasuk String) tidak diperbolehkan.
- Kedudukan statement pada default sama dengan kedudukan else pada if-else.

```
public class Switch {  
    public static void main(String args[]) {  
        int i = 2;  
        switch (i) {  
            case 1 : i+=3;  
                break;  
            case 2 : i+=5;  
                break;  
            default: i+=10;  
        }  
        System.out.println(i);  
    }  
}
```

Output

7



```
public class Switch {  
    public static void main(String args[]) {  
        int i = 1;  
        switch (i) {  
            case 1 : i+=3;  
            case 2 : i+=5;  
            default: i+=10;  
        }  
        System.out.println(i);  
    }  
}
```

Output

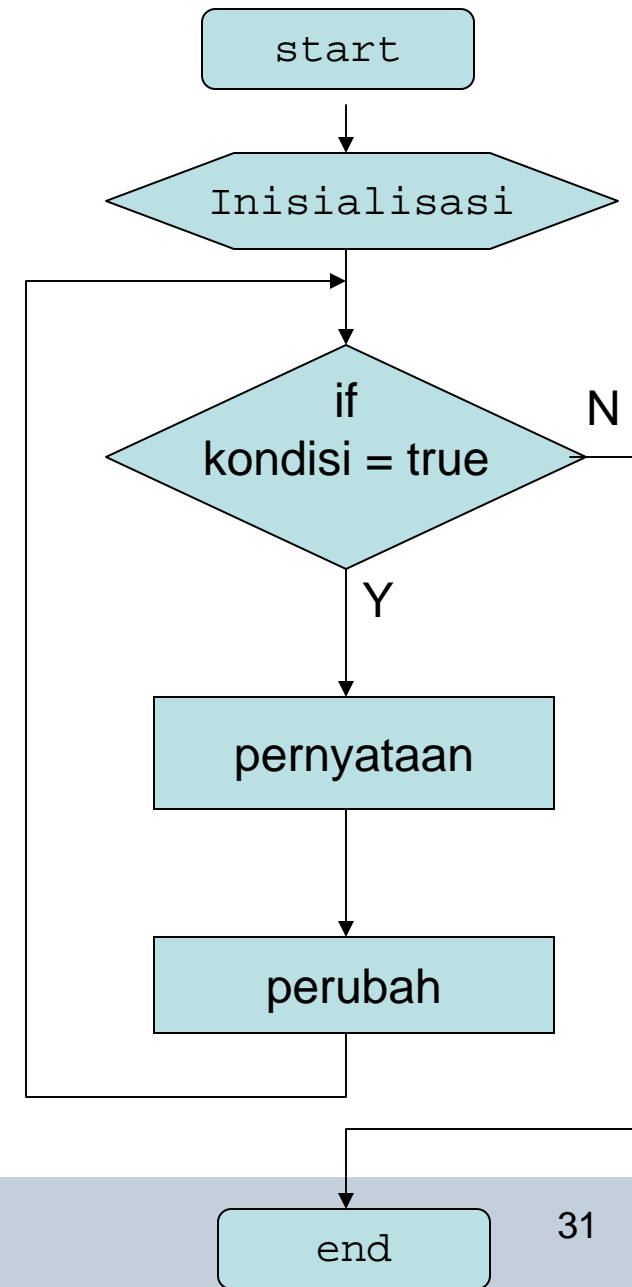
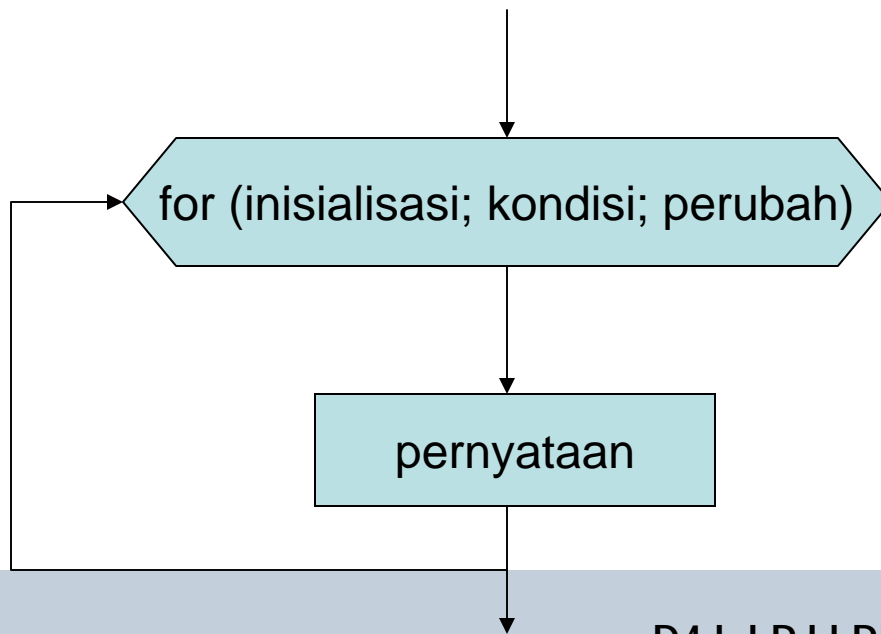
19

Perulangan

- for
- while
- do-while
- nested loop
- continue
- break

for

```
for (inisialisasi; kondisi; perubah) {  
    pernyataan;  
}
```

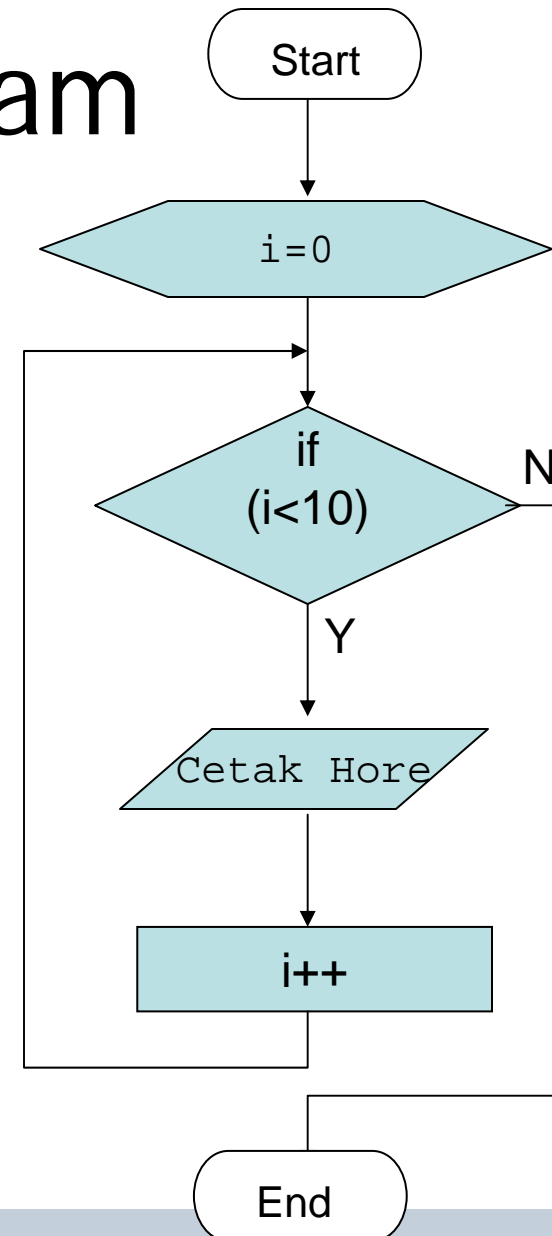


Contoh Program

```
for (int i=0; i<10; i++) {  
    System.out.println("Hore !!");  
}
```

Output

```
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!  
Hore !!
```




```

1 public class for1{
2     public static void main(String args[]){
3         for (int i=0; i<10; i++) {
4             System.out.println("Hore !!");
5         }
6         System.out.println("Finally");
7     }
8 }

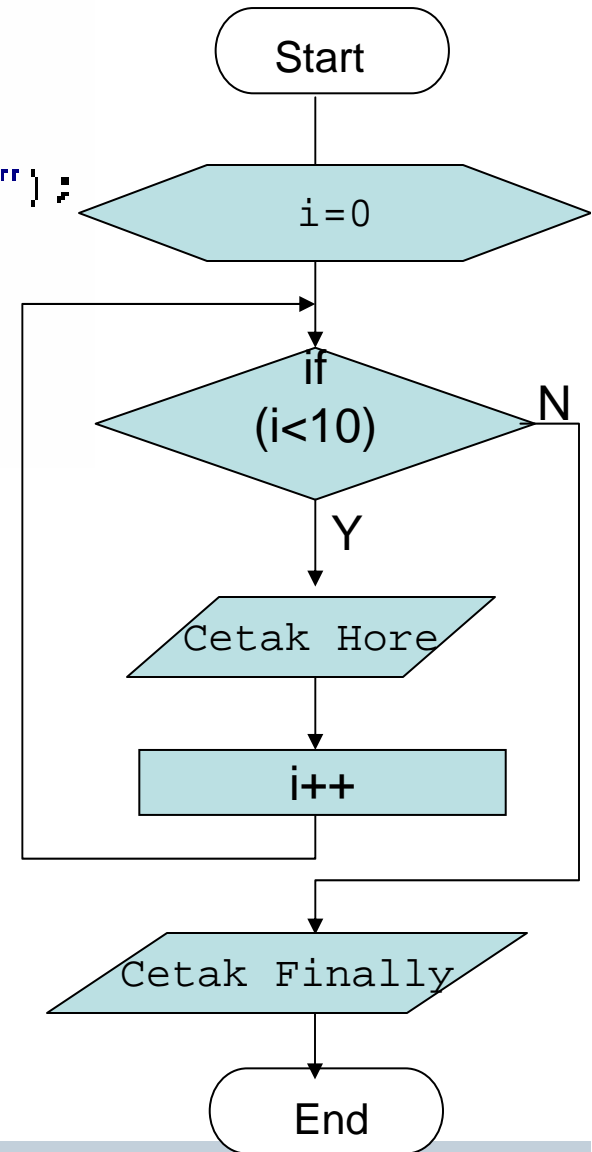
```

Output

```

Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Hore !!
Finally

```



for

- Di Java diperbolehkan menggunakan tanda koma untuk pemisah di for
- Contoh:

```
for (i=0, j = 0; j<10; i++, j++){ }
```



Contoh Program

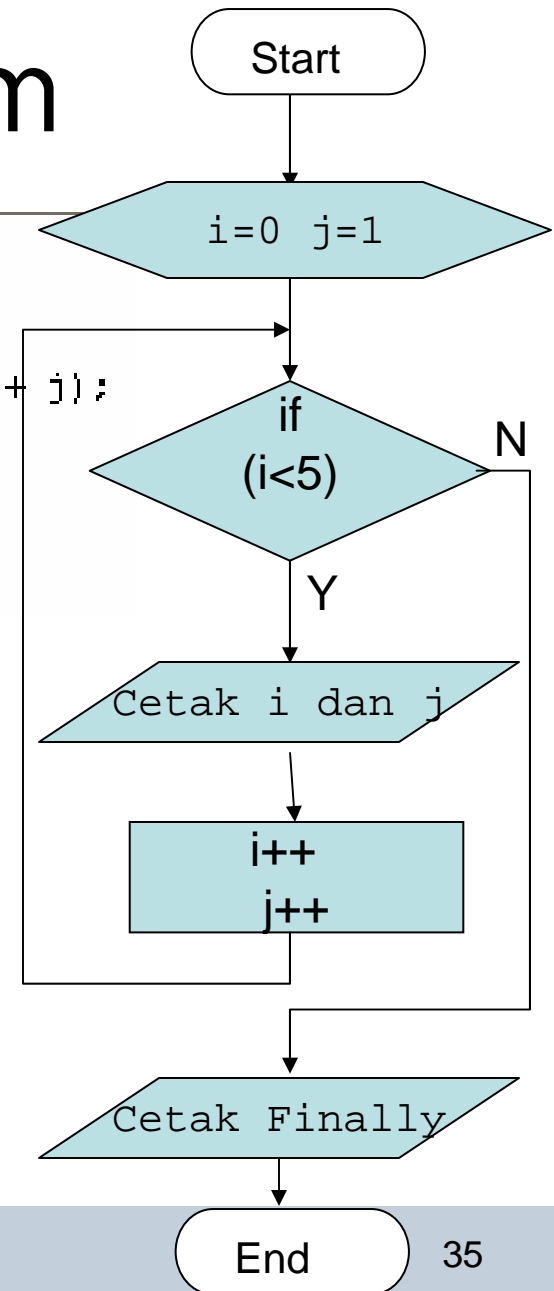
```

1 public class for2{
2     public static void main(String args[]){
3         for (int i=0, j=1; i<5 ; i++, j++) {
4             System.out.println("i = " + i + "    j = " + j);
5         }
6         System.out.println("Finally");
7     }
8 }
    
```

Output

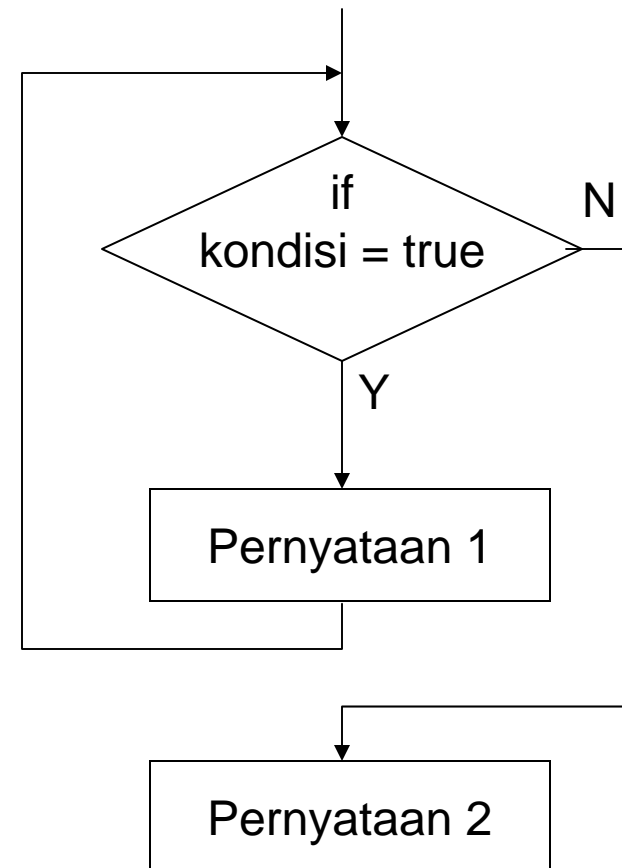
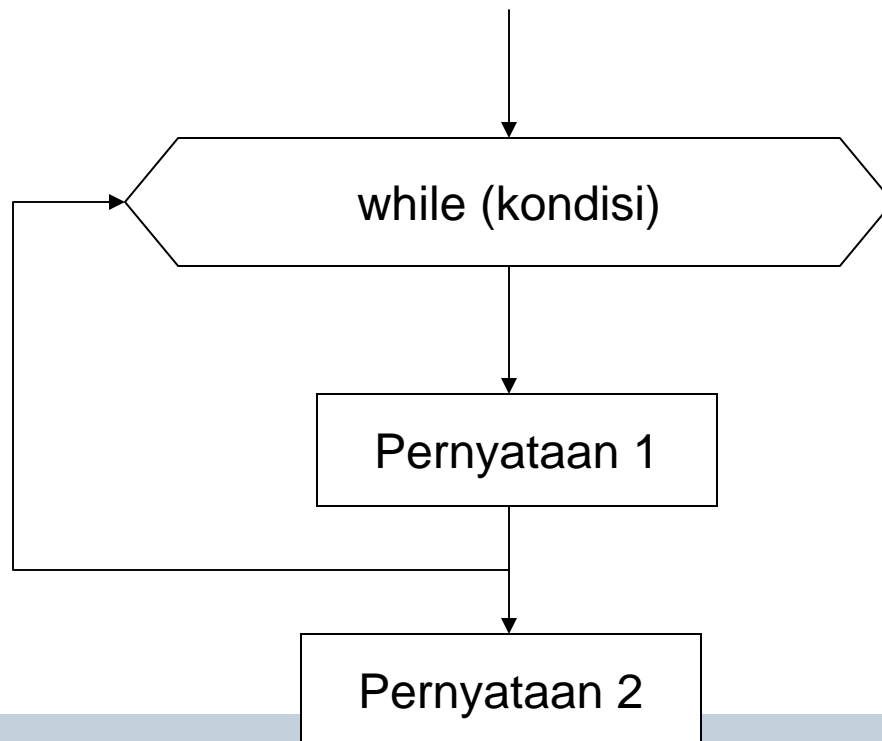
```

i = 0    j = 1
i = 1    j = 2
i = 2    j = 3
i = 3    j = 4
i = 4    j = 5
Finally
    
```



while

```
while (kondisi) {  
    pernyataan 1;  
}  
pernyataan 2
```



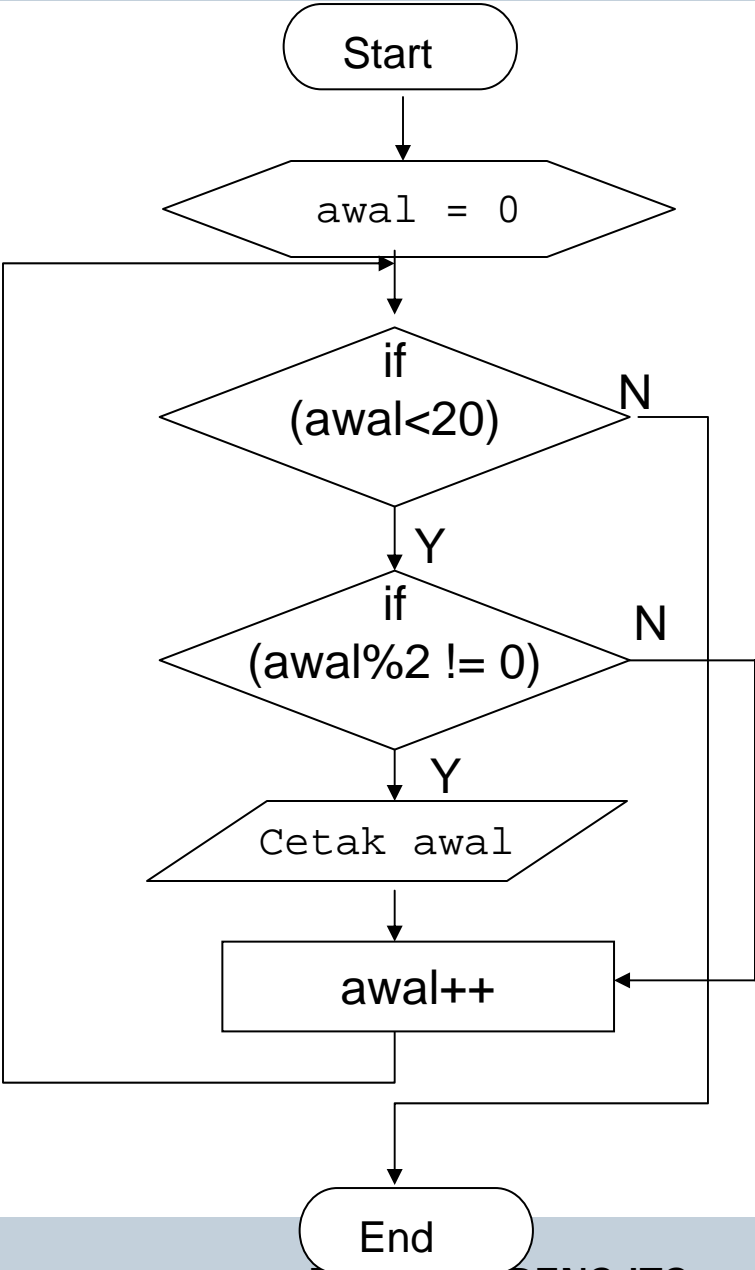
Contoh Program

- Menampilkan bilangan ganjil yang lebih kecil dari 20

```
1 public class while1{  
2     public static void main(String args[]){  
3         int awal;  
4         awal = 0 ;  
5         while (awal <=20){  
6             if (awal % 2 !=0)  
7                 System.out.print(awal+ " ");  
8                 awal++;  
9         }  
10    }  
11 }
```

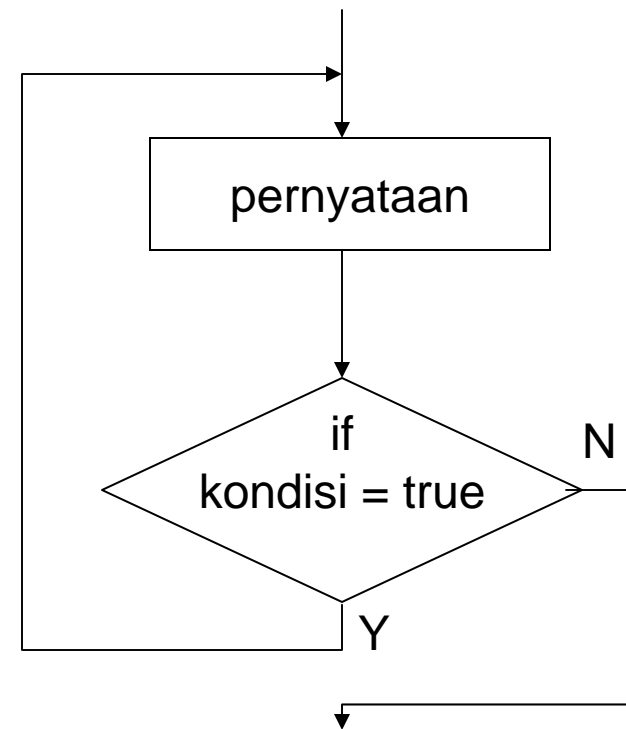
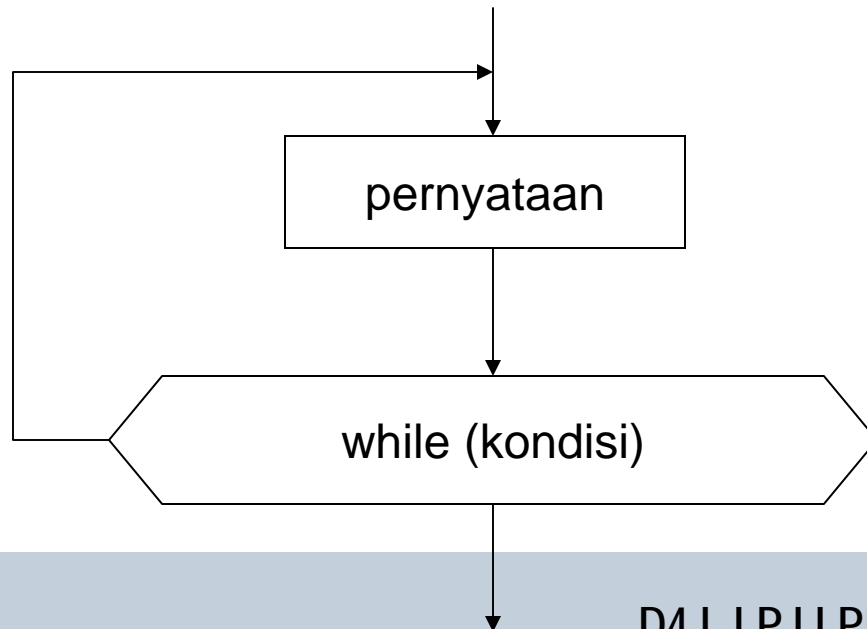
Output

1 3 5 7 9 11 13 15 17 19



do-while

```
do {  
    pernyataan;  
} while (kondisi);
```



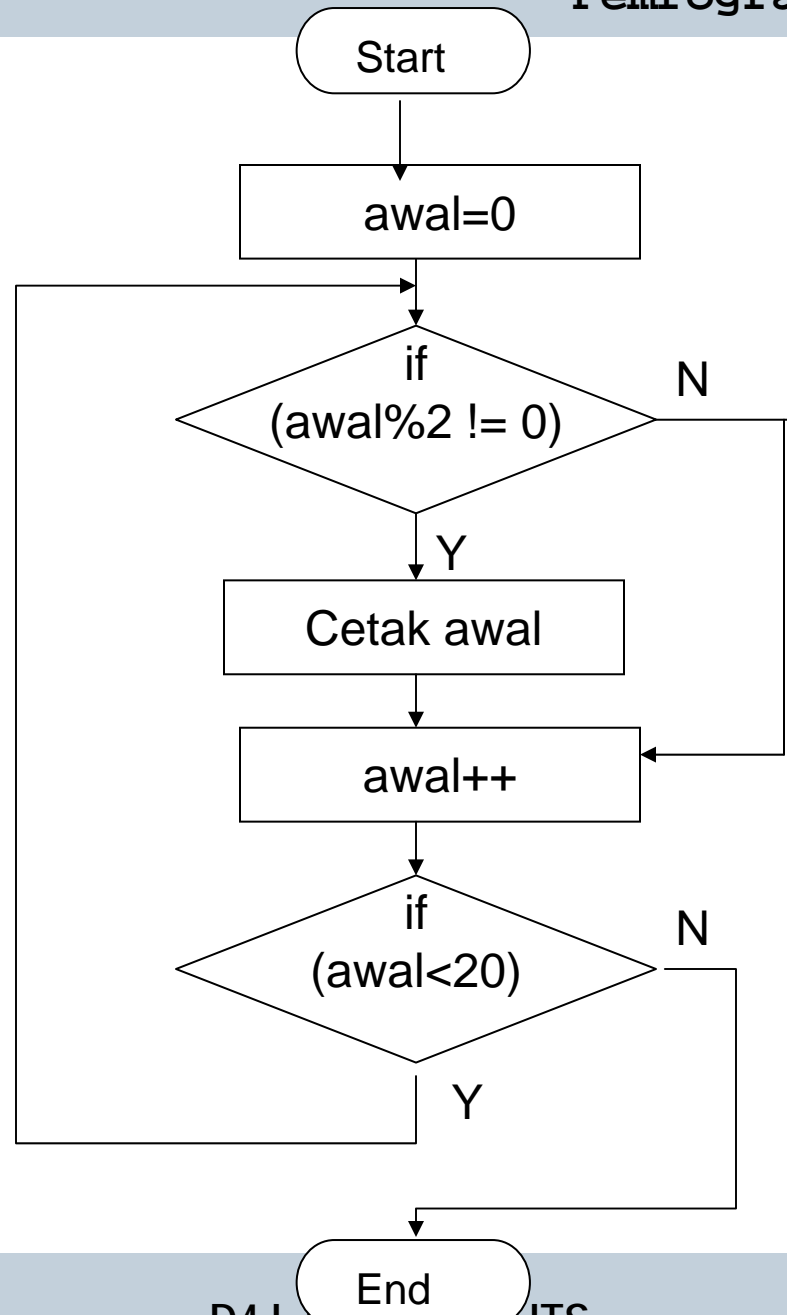
Contoh Program

- Menampilkan bilangan ganjil yang lebih kecil dari 20

```
1 public class while2{
2     public static void main(String args[]){
3         int awal;
4         awal = 0 ;
5         do{
6             if (awal % 2 !=0)
7                 System.out.print(awal+ " ");
8                 awal++;
9         }while(awal <=20);
10    }
11 }
```

Output

1 3 5 7 9 11 13 15 17 19



Nested loop

- Dalam suatu loop bisa terkandung loop yang lain, sehingga jika loop tersebut diulangi n kali dan loop yang didalamnya juga dilakukan m kali maka pernyataan yang berada didalam nested loop sebanyak $n \times m$.
- Loop yang terletak di dalam loop biasa disebut dengan loop di dalam loop (nested loop)

Output ?

- `public class loop1 {`
- `public static void main(String args[]){`
- `for(int i=0;i<3;i++){`
- `for(int j=0;j<5;j++){`
- `System.out.println("i : " + i + " j : " + j);`
- `}`
- `}`
- `}`
- `}`

Output

- $i:0j:0$
- $i:0j:1$
- $i:0j:2$
- $i:0j:3$
- $i:0j:4$
- $i:1j:0$
- $i:1j:1$
- $i:1j:2$
- $i:1j:3$
- $i:1j:4$
- $i:2j:0$
- $i:2j:1$
- $i:2j:2$
- $i:2j:3$
- $i:2j:4$

Contoh Program

```
• public class loop2 {  
•     public static void main(String args[]){  
•         for(int i=1;i<=3;i++){  
•             for(int j=1;j<=3;j++){  
•                 System.out.print(i*j + "  ");  
•             }  
•             System.out.println();  
•         }  
•     }  
• }
```

1	2	3
2	4	6
3	6	9

Special Loop Control

- `break [label];`
- `continue [label];`
- `label : statement;` (statement ini berupa loop)

Special Loop Control

- **break** digunakan untuk keluar (“prematurely exit”) dari switch statements, loop statements, dan labeled blocks.
- **continue** digunakan untuk meneruskan (“skip over and jump) ke akhir dari loop body, dan kembali ke loop control statement.
- **label** digunakan untuk mengidentifikasi statement lain dimana statement lain ini meminta supaya block statement pada label ini dikerjakan.

Special Loop Flow Control

The break statement:

```
do {  
    statement;  
    if (condition is true) {  
        break;  
    }  
    statement;  
} while (boolean expression);
```


Contoh Program

- Menampilkan bilangan ganjil yang lebih kecil dari 20. Jika bertemu dengan angka 15 maka break dari proses looping

```
1 public class Break1{
2     public static void main(String args[]){
3         int awal;
4         awal = 0 ;
5         while (awal < 20){
6             awal++;
7             if (awal == 15 )
8                 break ;
9             if (awal % 2 != 0)
10                System.out.print(awal+ " ");
11        }
12    }
13 }
14 }
```

Output

1 3 5 7 9 11 13

Special Loop Flow Control

The continue statement:

```
do {  
    statement;  
    if (boolean expression) {  
        continue;  
    }  
    statement;  
} while (boolean expression);
```

Contoh Program

- Menampilkan bilangan ganjil yang lebih kecil dari 20. Angka 15 tidak ditampilkan

```
1 public class Continue{
2     public static void main(String args[]){
3         int awal;
4         awal = 0 ;
5         while (awal < 20){
6             awal++;
7             if (awal == 15){
8                 continue ;
9             }
10            if (awal % 2 != 0)
11                System.out.print(awal+ " ");
12
13        }
14    }
15 }
```

Output

1 3 5 7 9 11 13 17 19

Special Loop Flow Control

Using break with labels:

```
outer:  
  do {  
    statement;  
    do {  
      statement;  
      if (boolean expression) {  
        break outer;  
      }  
      statement;  
    } while (boolean expression);  
    statement;  
  } while (boolean expression);
```

Contoh Program

```
1 public class BreakOuter{
2     public static void main(String args[]){
3         outer:
4         for (int i=0; i<3; i++){
5             for (int j=0; j<2; j++){
6                 System.out.println("i = " + i + "   j = " + j);
7                 break outer;
8             }
9         }
10        System.out.println("Good-Bye");
11    }
12 }
```

Output

```
i = 0   j = 0
Good-Bye
```

Special Loop Flow Control

Using continue with labels:

```
test:
  do {
    statement;
    do {
      statement;
      if (condition is true) {
        continue test;
      }
      statement;
    } while (condition is true);
    statement;
  } while (condition is true);
```

Contoh Program

```
1 public class ContinueOuter{
2     public static void main(String args[]){
3         outer:
4         for (int i=0; i<3; i++){
5             for (int j=0; j<2; j++){
6                 System.out.println("i = " + i + "   j = " + j);
7                 continue outer;
8             }
9         }
10        System.out.println("Good-Bye");
11    }
12 }
```

Output

```
i = 0   j = 0
i = 1   j = 0
i = 2   j = 0
Good-Bye
```

Soal

```
1. for (int i = 0; i < 2; i++) {  
2.   for (int j = 0; j < 3; j++) {  
3.     if (i == j) {  
4.       continue;  
5.     }  
6.     System.out.println("i = " + i + " j = " + j);  
7.   }  
8. }
```

Which lines would be part of the output? (Choose all that apply.)

- a i = 0 j = 0
- b i = 0 j = 1
- c i = 0 j = 2
- d i = 1 j = 0
- e i = 1 j = 1
- f i = 1 j = 2

Soal

```
1. outer: for (int i = 0; i < 2; i++) {  
2.     for (int j = 0; j < 3; j++) {  
3.         if (i == j) {  
4.             continue outer;  
5.         }  
6.         System.out.println("i = " + i + " j = " + j);  
7.     }  
8. }
```

Which lines would be part of the output? (Choose all that apply.)

- a i = 0 j = 0
- b i = 0 j = 1
- c i = 0 j = 2
- d i = 1 j = 0
- e i = 1 j = 1
- f i = 1 j = 2