Temu03

Struktur Kendali & array

Ali Tarmuji, S.T., M.Cs.
alitarmuji@tif.uad.ac.id

Bahan Diskusi

- Blok program
- Struktur Kendali
  - Percabangan (kondisi)
  - Struktur Perulangan (looping)
- Array (larik) :
  - Tipe data array
  - Memanipulasi array
  - Array 2 dimensi

Blok Program

Menjelaskan sintaks dan blok program dalam pemrograman bahasa Java

Blok program

- Sebelum belajar percabangan, kita harus terlebih dahulu mengenal blok.
- Blok menandakan kelompok perintah dalam satu bagian kesatuan yang biasanya ditandai dengan simbol khusus
- Blok di Java merupakan area kode yang diapit oleh kurung kurawal buka { dan diakhiri dengan kurung kurawal tutup }
- Blok dapat dibuat di dalam blok lain
- Sebuah variabel di blok terluar, dapat diakses dari blok di dalamnya →scope variable global
- Namun sebaliknya, variabel di blok dalam, **tidak dapat** diakses dari blok luarnya →scope variabel lokal

Struktur Kendali
-Percabangan-

Menjelaskan sintaks dan penggunaan struktur percabangan dalam pemrograman Java

Percabangan IF Tunggal

- Sintaks umum:

```
If (<kondisi>) {
    <isi perintah jika kondisi benar>
}
```

- Kondisi biasanya menggunakan operator relasional

```
<       // less than
<=      // less than or equal to
==      // equal to
!=      // not equal to
>       // greater than
>=      // greater than or equal to
```

Contoh IF Tunggal

```
2
3       import java.util.Scanner;
4
5       public class Main {
6           public static void main(String args[])
7           {   Scanner scanner = new Scanner(System.in);
8               System.out.print("Enter test score: ");
9               int testScore = scanner.nextInt();
10              if (testScore < 70)
11                  System.out.println("You did not pass");
12          }
13      }
14
```

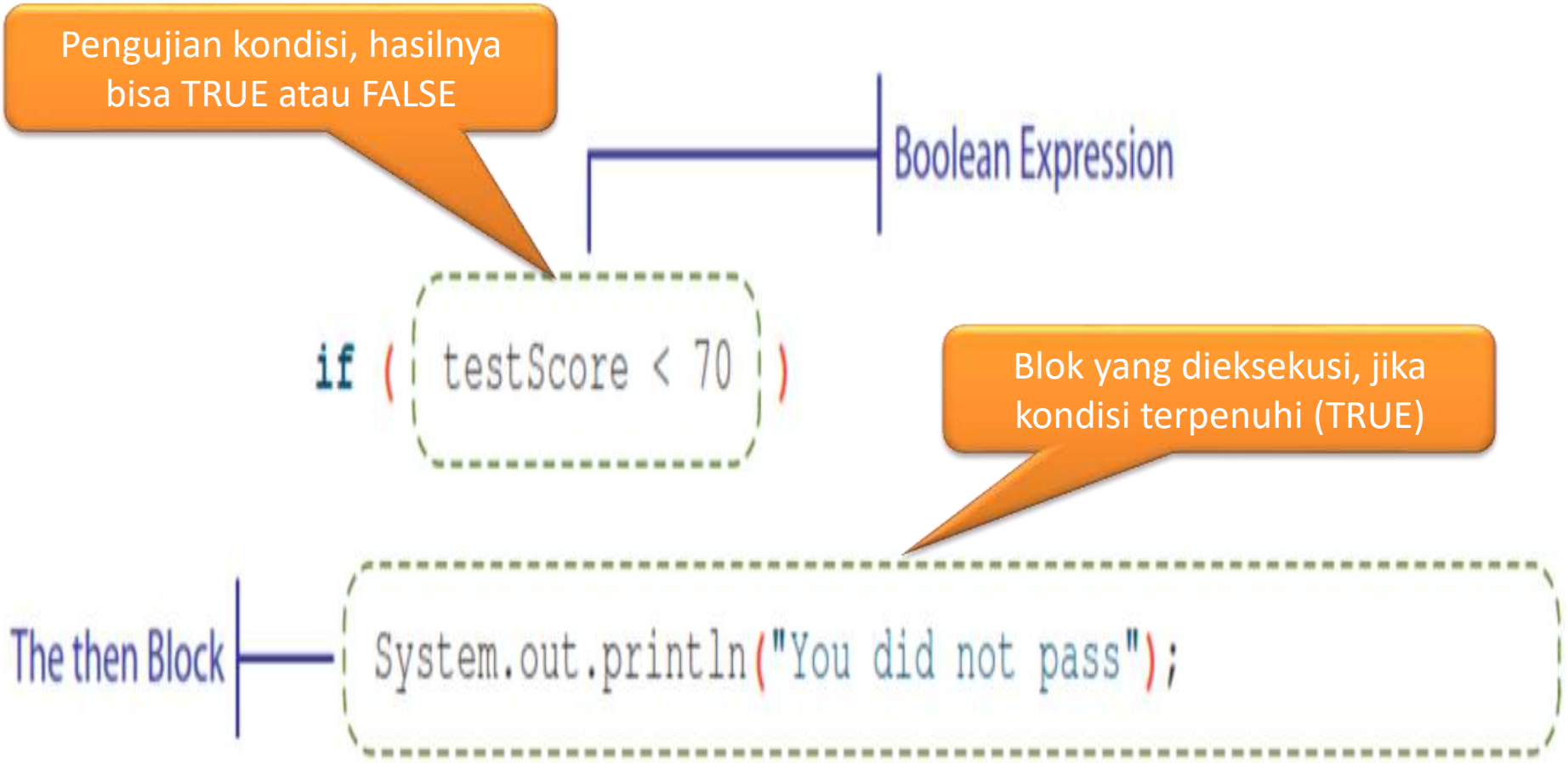
Main main()

Run: Main

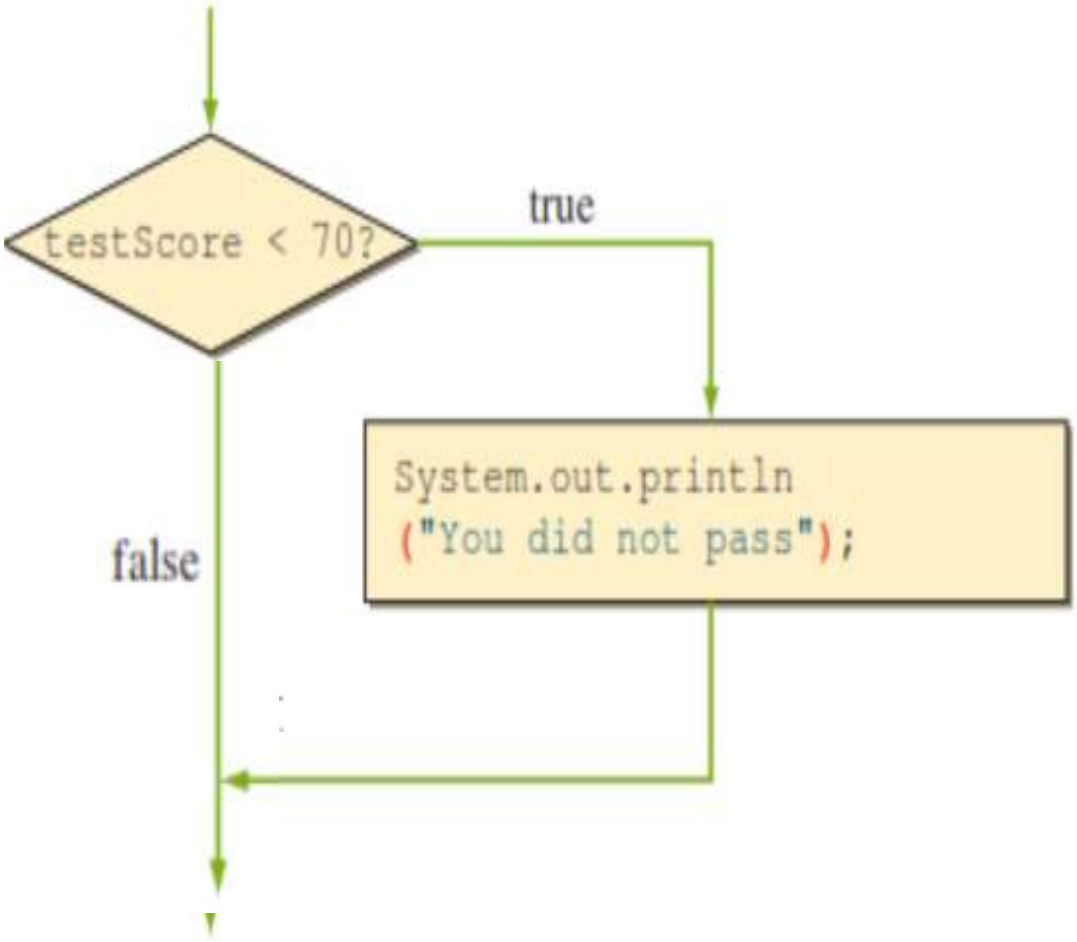
```
Enter test score: 65
You did not pass

Process finished with exit code 0
```

Penjelasan


Pengujian kondisi, hasilnya bisa TRUE atau FALSE
Boolean Expression
if (testScore < 70)
Blok yang dieksekusi, jika kondisi terpenuhi (TRUE)
The then Block
System.out.println("You did not pass");

Flowchart IF



Percabangan IF ... ELSE

- Sintaks umum:

```
If (<kondisi>) {
    <isi perintah jika kondisi benar>
}
```

Contoh penggunaan IF...ELSE

```
3       import java.util.Scanner;
4
5       public class Main {
6           public static void main(String args[])
7           {   Scanner scanner = new Scanner(System.in);
8               System.out.print("Enter test score: ");
9               int testScore = scanner.nextInt();
10              if (testScore < 70)
11                  System.out.println("You did not pass");
12              else
13                  System.out.println("You did pass");
14          }
```

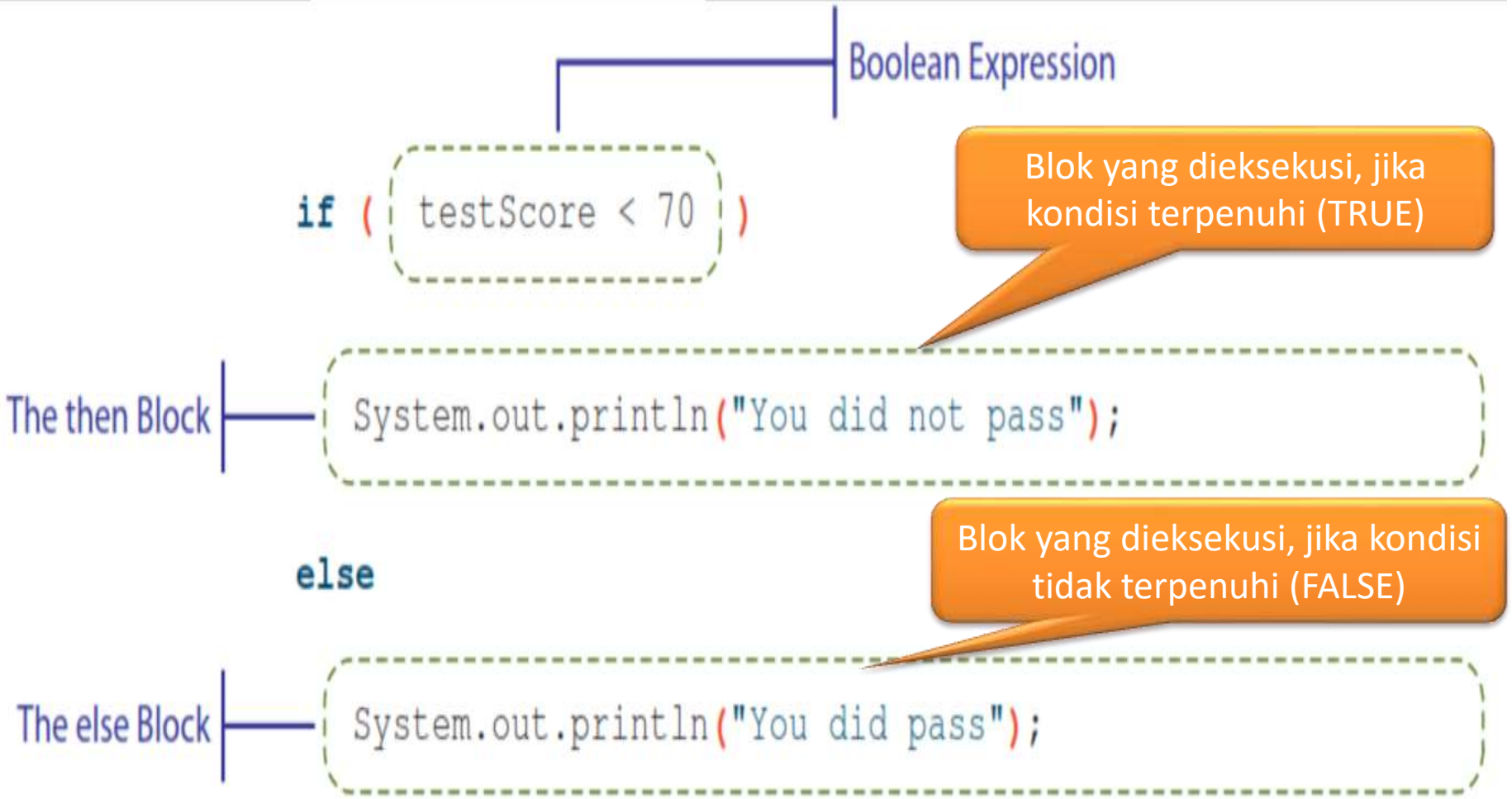
Main > main()

Run: Main

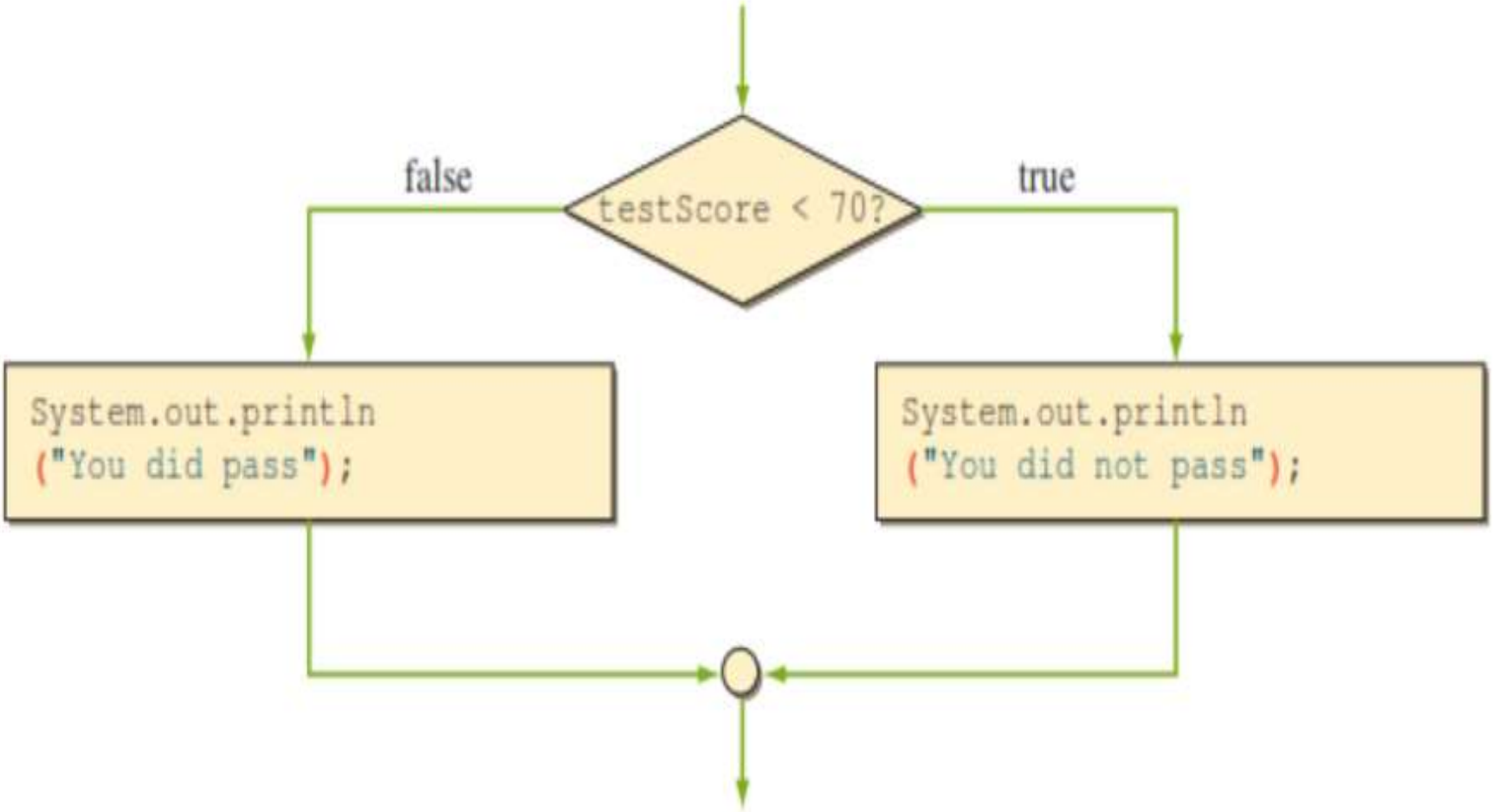
```
Enter test score: 75
You did pass

Process finished with exit code 0
```

Penjelasan


Boolean Expression
if (testScore < 70)
Blok yang dieksekusi, jika kondisi terpenuhi (TRUE)
The then Block
System.out.println("You did not pass");
else
Blok yang dieksekusi, jika kondisi tidak terpenuhi (FALSE)
The else Block
System.out.println("You did pass");

Flowchart IF...ELSE



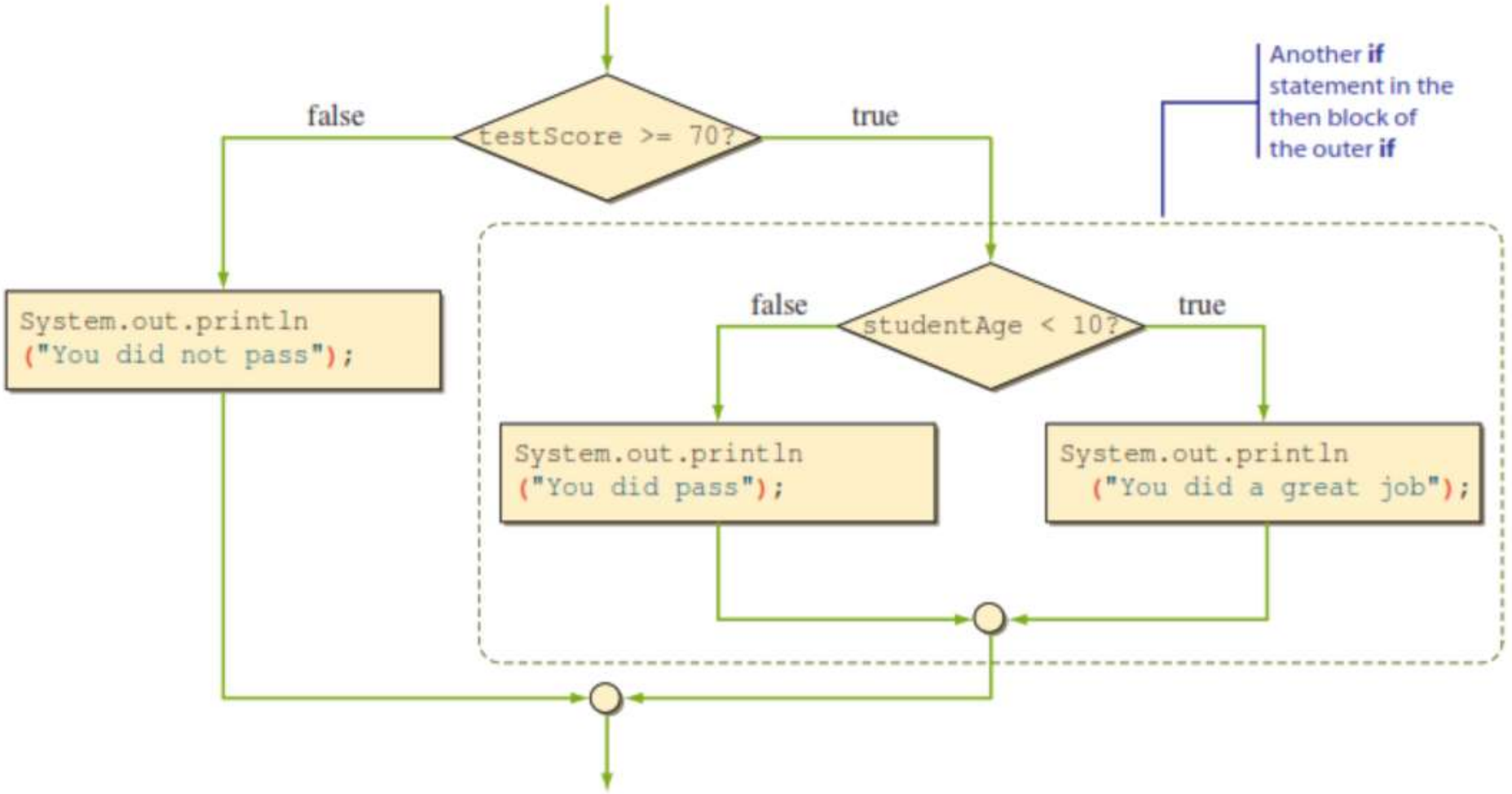
Percabangan Bersarang (Nested IF)

```
if (testScore >= 70) {
   if (studentAge < 10) {
      System.out.println("You did a great job");
   } else {

      System.out.println("You did pass");//test score >= 70
   }                                      //and age >= 10
} else { //test score < 70

   System.out.println("You did not pass");
}
```

Flowchart Nested IF



Variasi penulisan Nested IF

```
if (testScore >= 70 && studentAge < 10) {
    System.out.println("You did a great job");
} else {
    //either testScore < 70 OR studentAge >= 10

    if (testScore >= 70) {
       System.out.println("You did pass");
    } else {
       System.out.println("You did not pass");
    }
}
```

Variasi penulisan Nested IF

```
if (num1 < 0)
    if (num2 < 0)
        if (num3 < 0)
            negativeCount = 3; //all three are negative
        else
            negativeCount = 2; //num1 and num2 are negative
    else
        if (num3 < 0)
            negativeCount = 2; //num1 and num3 are negative
        else
            negativeCount = 1; //num1 is negative
else
    if (num2 < 0)
        if (num3 < 0)
            negativeCount = 2; //num2 and num3 are negative
        else
            negativeCount = 1; //num2 is negative
    else
        if (num3 < 0)
            negativeCount = 1; //num3 is negative
        else
            negativeCount = 0; //no negative numbers
```