SELECTION STATEMENTS (1)

if

if…else

CSC 111

Outline

1. Introduction

2. One-way selection: The if statement

    2.1 With a single statement

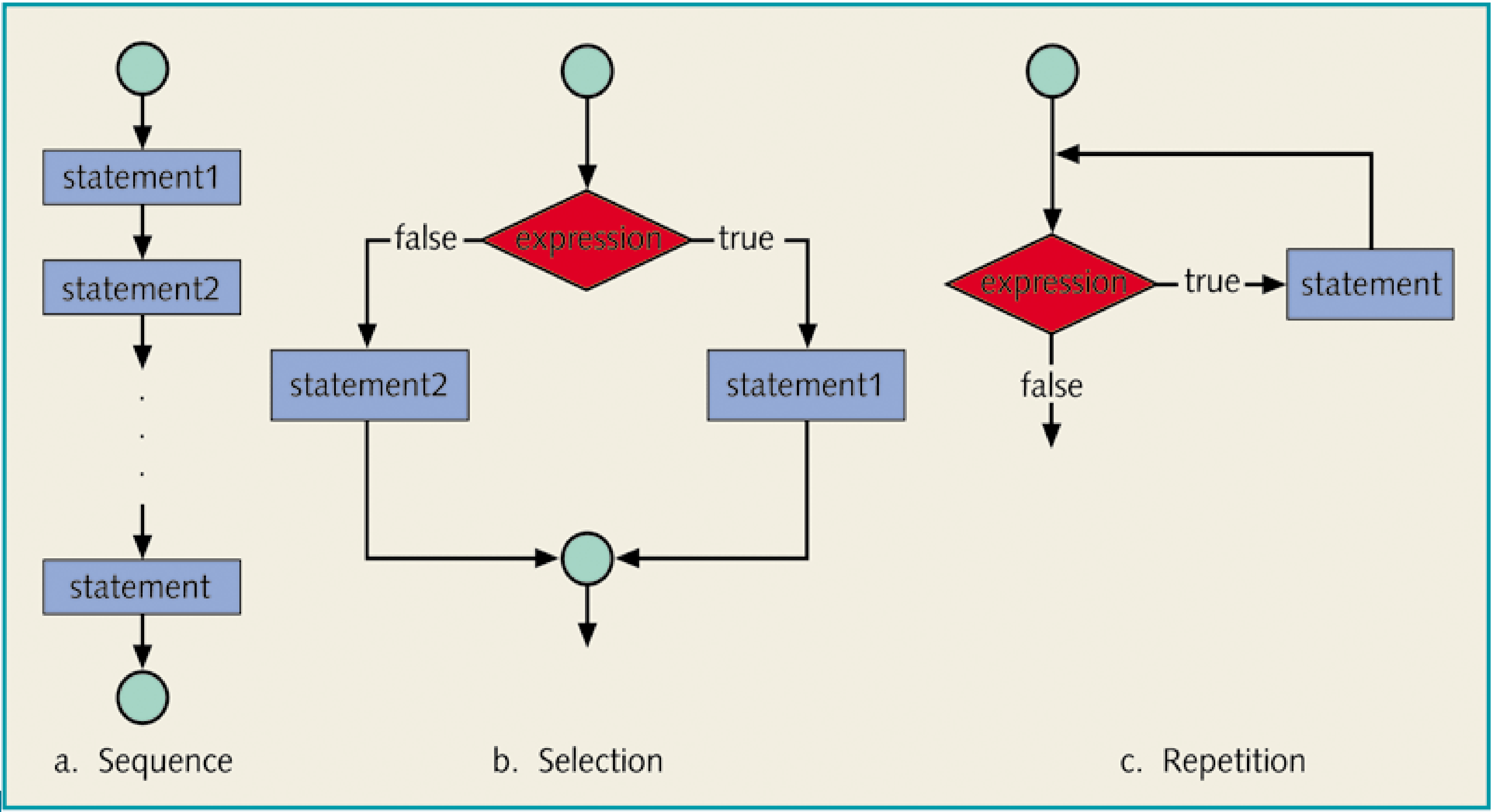
    2.2 With a block of statements

3. Two-way selection: The if…else statement

    3.1 With a single statement

    3.2 With a block of statements

Selection


statement1
statement2
statement
false
expression
true
statement2
statement1
expression
true
statement
false
a. Sequence
b. Selection
c. Repetition

1. Introduction

- By default, the flow of a program is sequential.
- Selection statements alters the flow of execution depending on one or more specific conditions.
- The condition(s) is/are specified by the programmer.
- In Java, the selection statements include:
  - The if statement
    - With a single statement
    - With a block of statements
  - The if…else statement
    - With a single statement
    - With a block of statements
  - The nested if
  - The conditional operator ?
  - The switch statement

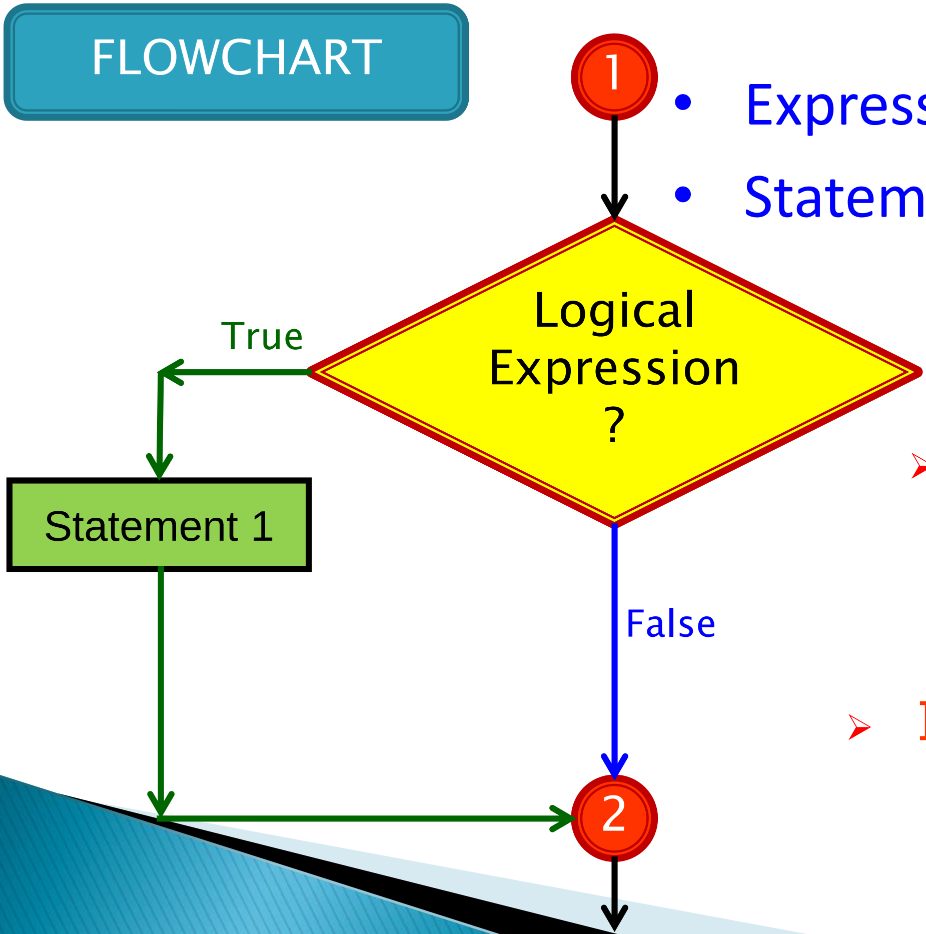
2. The if Statement

With a single statement

SYNTAX

```
if (logical expression) statement1;
```

FLOWCHART



- Expression referred to as **decision maker**.
- Statement referred to as **action statement**.

➢ If the logical expression is true:

  ○ Statement 1 is executed

  ○ Execution continues at point 2

➢ If the logical expression is false:

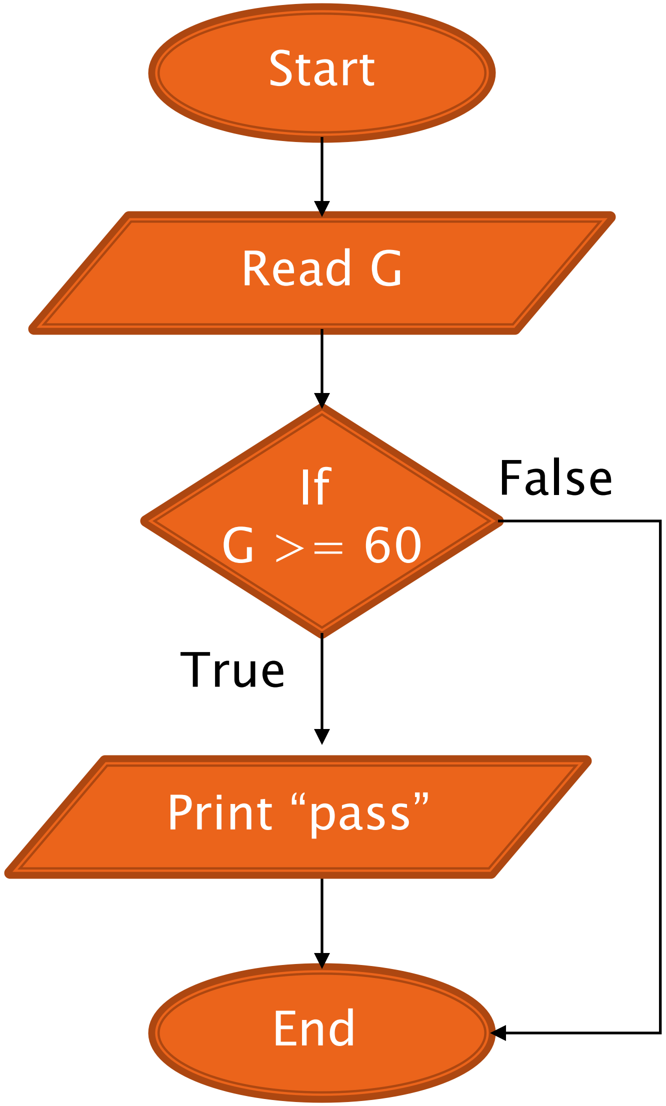
  ○ Statement 1 is NOT executed

  ○ Execution continues at point 2

Write a program that read the grade of the student and if it is more than or equal 60; print “Pass”

Algorithm

- Start
- Read the grade G
- If G more than or equal to 60
  - Print “Pass”
- End



Code

```
import java.util.*;
public class grade
{
   static Scanner console = new Scanner (System.in);
   public static void main(String[] args)
   {
      double grade;
      System.out.println("Enter Grade");
      grade = console.nextDouble();
      if (grade >= 60)
System.out.println(“PASS“);
   }
}
```

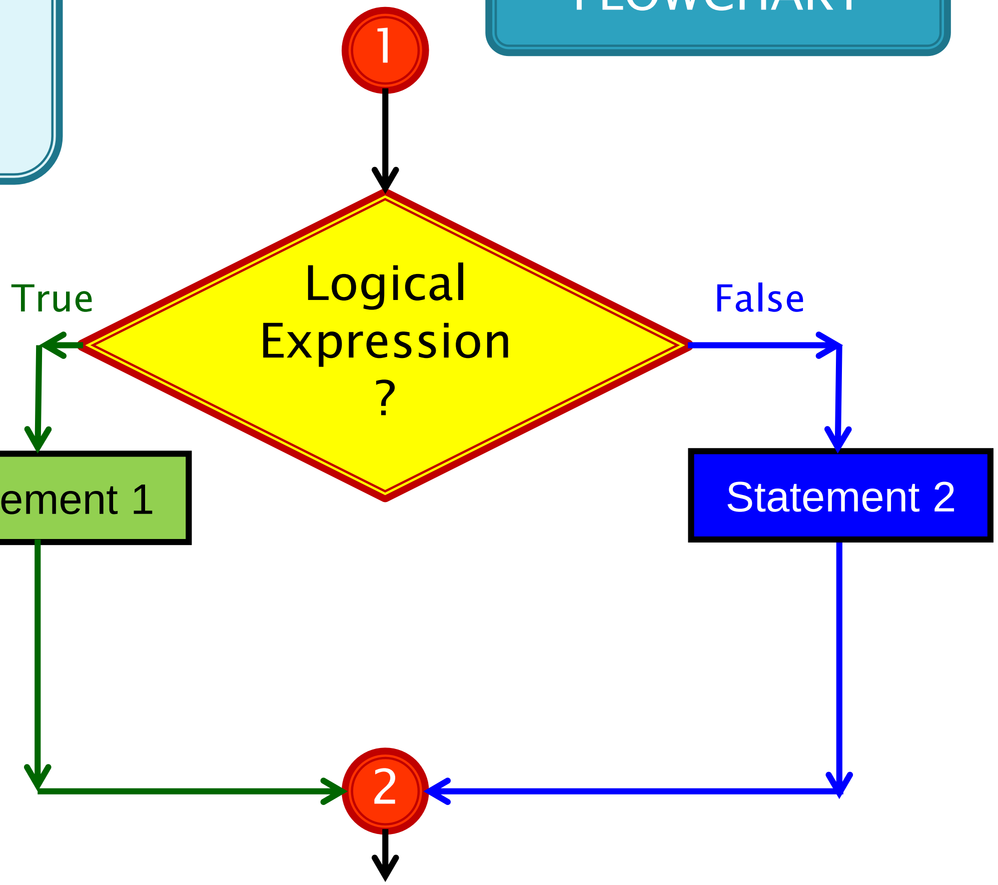
3.1 The if...else Statement

With a single statement

SYNTAX

```
if (logical expression)
   Statement1;
else
   Statement2;
```

FLOWCHART



- If the logical expression is true:
  - ○ Statement 1 is executed
  - ○ Execution continues at point 2
- If the logical expression is false:
  - ○ Statement 2 is executed
  - ○ Execution continues at point 2

3.1 The if...else Statement

With a single statement – PROGRAM 2: ANALYSIS

Write a program that read two numbers and print the largest one

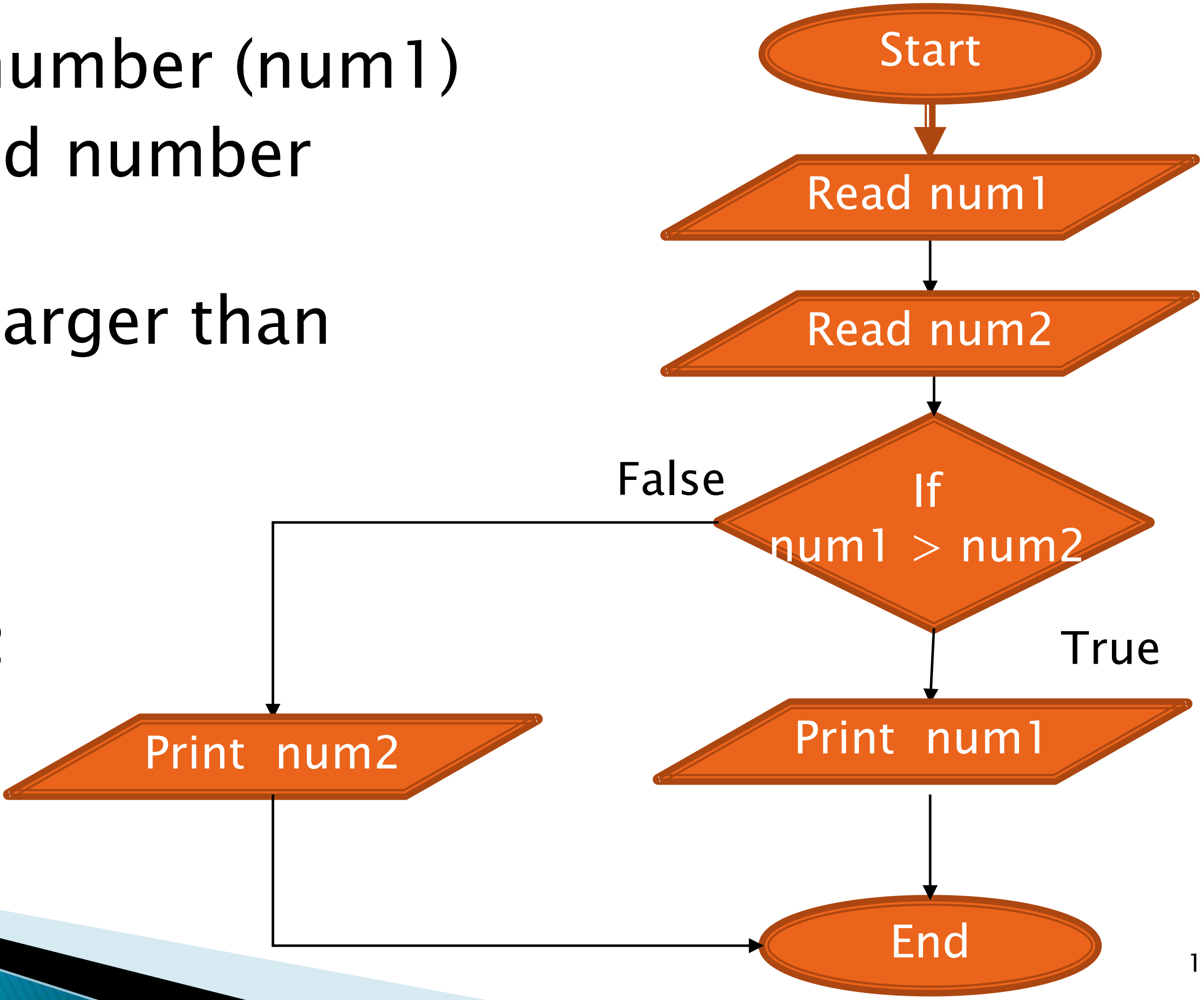
INPUT	Number1 : (variable: num1, type: int) Number2 : (variable: num2, type: int)
OUTPUT	Largest number

PROCESS	if (num1 > num2) print num1;
	if (num2 > num1) print num2;

3.1 The if...else Statement

With a single statement – PROGRAM 3: Algorithm – Flowchart

- Start
- Read first number (num1)
- Read second number (num2)
- If num1 is larger than num2
  - Print num1
- else
  - Print num2



3.1 The if...else Statement

With a single statement – PROGRAM 3: CODE

```
// import necessary libraries
import java.util.*;                    //contains the class Scanner
public class ifElseStatement1
{
  // instantiate the object console from the class Scanner
  static Scanner console = new Scanner (System.in);
  public static void main (String[] args)
    {
      // Declaration section: to declare needed variables
          int num1, num2;
      // Input section: to enter values of used variables
          System.out.println (“Enter number1”);   //prompt
          num1 = console.nextInt();
           System.out.println (“Enter number2”);   //prompt
          num2 = console.nextInt();
      // Processing section: processing statements
          if (num1 > num2)
           System.out.println (num1);
           else
             System.out.println (num2);
      // Output section: display program output
              } // end main
} // end class
```

Two-Way Selection

Example 4-14

//The following statement shows an example of a syntax error… WHY ?

```
if (hours > 40.0) ;                              //Line 1
    wages = 40.0 * rate + 1.5 * rate *(hours - 40.0); //Line 2
else                                            //Line 3
    wages = hours * rate;                       //Line 4
```

- Because a semicolon follows the closing parenthesis of the `if` statement (Line 1), the `else` statement stands alone.
- The semicolon at the end of the `if` statement (see Line 1) ends the `if` statement.
- The statement at Line 2 separates the else clause from the `if` statement. That is, `else` is by itself.
- Because there is no separate `else` statement in Java, this code generates a syntax error.

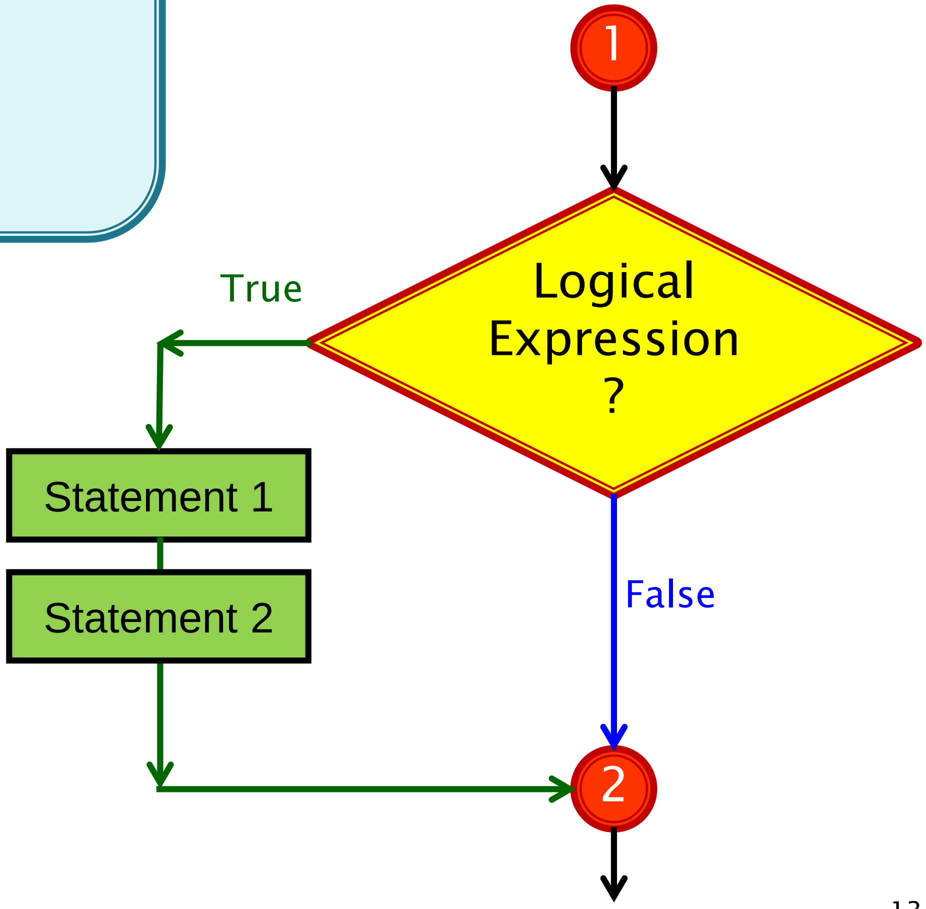
2.2 The if Statement

With a block of statements

SYNTAX

```
if (logical expression)
{
  statement 1;
  statement 2;
  …
  statement n;
}
```

FLOWCHART

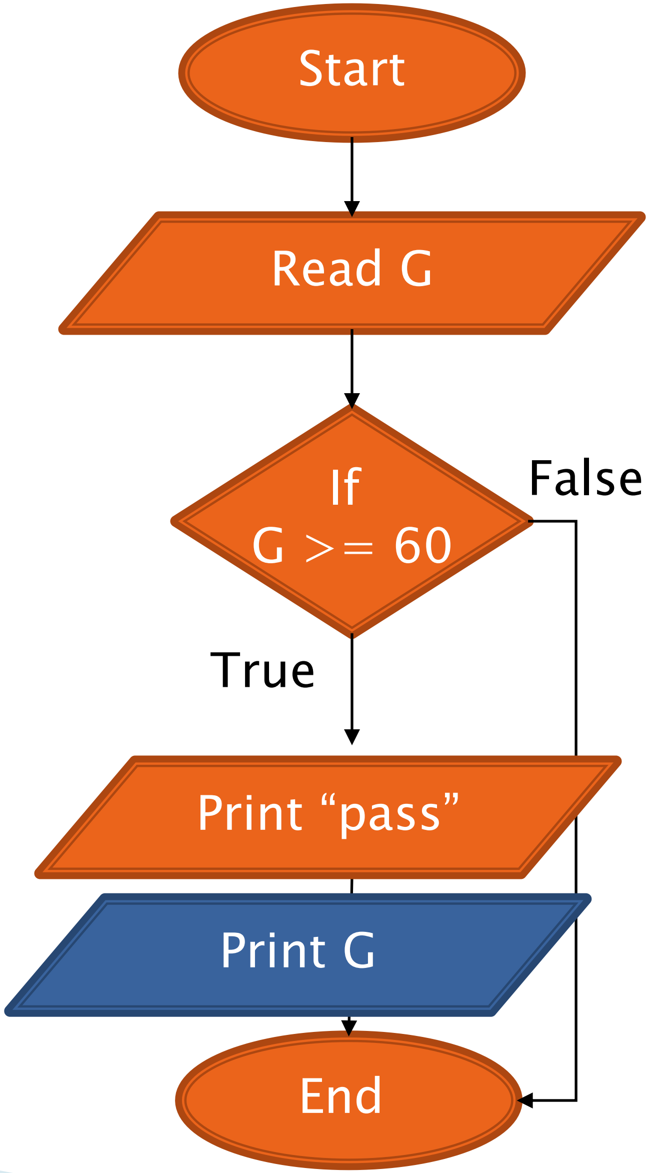


- If the logical expression is true:
  - Statements 1 & 2 are executed
  - Execution continues at point 2
- If the logical expression is false:
  - Execution continues at point 2

Write a program that read the grade of the student and if it is more than or equal 60; print “Pass” and the grade

Algorithm

- Start
- Read the grade G
- If G more than or equal to 60
  - Print “Pass”
  - Print G
- End



Code

```
import java.util.*;
public class grade
{ // Class start
   static Scanner console = new Scanner (System.in);
   public static void main(String[] args)
   {   // method start
      double grade;
      System.out.println("Enter Grade");
      grade = console.nextDouble();
      if (grade >= 60)
      {  // if block  start
       System.out.println(“PASS“);
       System.out.println(grade);
      } // if block  end
   } // method end
} // Class end
```

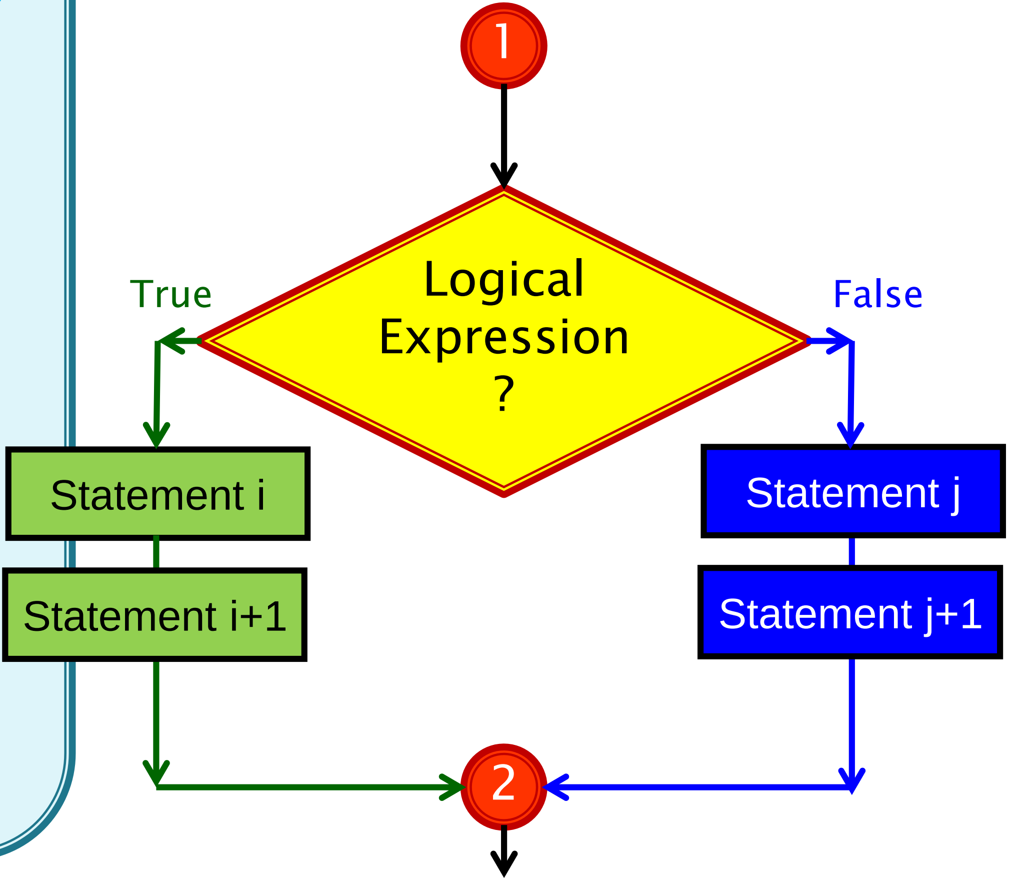
3.2 The if…else Statement

With a block of statements

SYNTAX

```
if (logical expression)
   {
    Statement i;
    Statement i+1;
    …
    Statement m;
   }
else
   {
     Statement j;
     Statement j+1;
     …
     Statement n;
   }
```

FLOWCHART



3.2 The if…else Statement

With a block of statements – PROGRAM 4: ANALYSIS

Write a program that assigns a special discount to an item according to its price, then calculates the net price as follows:
If the price is greater than 1,000 SR then the customer is given a VIP status and the item is given a 15% discount; otherwise, the item is given an 18% discount. The program should also display the customer status if VIP.

INPUT

Price of the item (variable: price, type: double)

OUTPUT

Discount (variable: discount, type: double)
Net Price (variable: netPrice, type: double)
Customer Status (variable: vip, type: boolean)

PROCESS

```
if (price > 1000) ➔          1) discount = 0.15
                              2) vip = true
if (price <= 1000) ➔         1) discount = 0.18
                              2) vip = false
netPrice = (1 – discount) * price
```

3.2 The if…else Statement

With a block of statements – PROGRAM 4: CODE

```
// import necessary libraries
import java.util.*;                   //contains the class Scanner
public class ifElseStatementN
{
  // instantiate the object console from the class Scanner
  static Scanner console = new Scanner (System.in);
  public static void main (String[] args)
    {
      // Declaration section: to declare needed variables
         double price, discount, netPrice;
         boolean vip;
      // Input section: to enter values of used variables
         System.out.println (“Enter the price of the item”);   //prompt
         price = console.nextDouble();
      // Processing section: processing statements
          if (price > 1000.0)                       // price is double
            {
             discount = 0.15;
             vip = true;
            } //end if(price > 1000.0)
          else
           {
             discount = 0.18;
             vip = false;
           } //end else if(price > 1000)
          netPrice = (1 – discount) * price;
      // Output section: display program output
         System.out.printf (“Discount = %.2f, Net price = %.2f, vip = %5s“, discount, netPrice, vip);
    } // end main
} // end class
```

2. CONDITIONAL OPERATOR ?

- Also known as the ternary operator.

SYNTAX

```
Variable =
 (LogicalExpression1) ? Expression2: Expression3;
```

•If expression1 = true, then the result of the condition is expression2. Otherwise, the result of the condition is expression3.

Example

```
int x = 3, y = 5, min;
min = (x <= y) ? x : y;
max = (x >= y) ? x: y;
System.out.println (min);
System.out.print (max);
```

Output

```
3
5_
```

2. CONDITIONAL OPERATOR ?

Example

```
int x = 3, y = 5, min;
min = (x <= y) ? x : y;
System.out.print (min);
```

- The above example is equivalent to the following code:

```
if (x <= y)
   min = x;
else
   min = y;
```

Self-Check Exercises

- Write a complete Java program that identifies the students whose grades are B.

- Write a complete Java program that reads two integer numbers num1 & num2, and performs a division if num2 is not equal to zero.

- Write a complete Java program that reads two integer numbers num1 & num2, and prints the smaller one.

- Write a complete Java program that identifies a negative number.

- Write a complete Java program that identifies an even number.