Chapter – 4

**4.1 Fill in the blanks in each of the following statements**:

a) All programs can be written in terms of three types of control structures **sequence, selection** and **repetition**.

b) The statement **if-else** is used to execute one action when a condition is true and

another when that condition is false.

c) Repeating a set of instructions a specific number of times is called **counter-controlled (or definite**) repetition.

d) When it’s not known in advance how many times a set of statements will be repeated,

a(n) **sentinel**, **signal**, **flag or dummy** value can be used to terminate the repetition.

e) The **sequence** structure is built into Java; by default, statements execute in the order

they appear.

f) Instance variables of types char, byte, short, int, long, float and double are all given

the value **0 (zero)** by default.

g) Java is a(n) **Strong typed** language; it requires all variables to have a type.

h) If the increment operator **prefixed** is to a variable, first the variable is incremented by

1, then its new value is used in the expression.

**4.2** State whether each of the following is *true* or *false*. If *false*, explain why.

a) An algorithm is a procedure for solving a problem in terms of the actions to execute and

the order in which they execute.

**True**.

b) A set of statements contained within a pair of parentheses is called a block.

**False. A set of statements contained within a pair of braces ({ and }) is called a**

**block**.

c) A selection statement specifies that an action is to be repeated while some condition remains

true.

**False. A repetition statement specifies that an action is to be repeated while some condition**

**remains true**.

d) A nested control statement appears in the body of another control statement.

**True**.

e) Java provides the arithmetic compound assignment operators +=, -=, \*=, /= and %= for

abbreviating assignment expressions.

**True**

f) The primitive types (boolean, char, byte, short, int, long, float and double) are portable

across only Windows platforms.

**False. The primitive types (boolean, char, byte, short, int,**

**long, float and double) are portable across all computer platforms that support Java.**

g) Specifying the order in which statements execute in a program is called program control.

**True**.

h) The unary cast operator (double) creates a temporary integer copy of its operand.

**False. The unary cast operator (double) creates a temporary floating-point copy of its operand.**

1. Instance variables of type boolean are given the value true by default.

**False. Instance variables of type boolean are given the value false by default**.

j) Pseudocode helps you think out a program before attempting to write it in a programming

language.

**True**

**4.3** Write four different Java statements that each add 1 to integer variable x.

Ans :

**x = x + 1;**

**x += 1;**

**++x;**

**x++;**

**4.4** Write Java statements to accomplish each of the following tasks:

1. Use one statement to assign the sum of x and y to z, then increment x by 1.

Ans : z = x++ + y;

1. Test whether variable count is greater than 10. If it is, print "Count is greater than 10".

Ans : **if ( count > 10 )**

**System.out.println( "Count is greater than 10" );**

c) Use one statement to decrement the variable x by 1, then subtract it from variable total

and store the result in variable total.

Ans : total -= --x;

d) Calculate the remainder after q is divided by divisor, and assign the result to q. Write

this statement in two different ways.

**Ans : q %= divisor;**

**q = q % divisor;**

**4.5 Write a Java statement to accomplish each of the following tasks**:

a) Declare variables sum and x to be of type int.

int sum;

int x;

1. Assign 1 to variable x.

**X=1;**

1. Assign 0 to variable sum.

**Sum = 0**;

1. Add variable x to variable sum, and assign the result to variable sum.

**Sum = sum+x or sum +=x**;

1. Print "The sum is: ", followed by the value of variable sum.

**System.out.printf( "The sum is: %d\n", sum );**

**4.6** Combine the statements that you wrote in Exercise 4.5 into a Java application that calculates

and prints the sum of the integers from 1 to 10. Use a while statement to loop through the

calculation and increment statements. The loop should terminate when the value of x becomes 11.

Ans :

1

**2**

**3** public class Calculate

**4** {

**5** public static void main( String[] args )

**6** {

**7** int sum;

**8** int x;

**9**

**10** x = 1;

**11** sum = 0;

**12**

**13** while ( x <= 10 )

**14** {

**15** sum += x;

**16** ++x;

**17** }

**18**

**19** System.out.printf( "The sum is: %d\n", sum );

**20** }

**21** }

**Output**:

|  |
| --- |
| The sum is: 55 |

**4.7** Determine the value of the variables in the statement product \*= x++; after the calculation

is performed. Assume that all variables are type int and initially have the value 5.

**Ans:** product = 25, x = 6

**4.8** Identify and correct the errors in each of the following sets of code:

a) while ( c <= 5 )

{

product \*= c;

++c;

Ans :

**Error: The closing right brace of the while statement’s body is missing.**

**Correction: Add a closing right brace after the statement ++c;**.

b) if ( gender == 1 )

System.out.println( "Woman" );

else;

System.out.println( "Man" );

Ans :

**Error: The semicolon after else results in a logic error. The second output statement**

**will always be executed.**

**Correction: Remove the semicolon after else.**

**4.9** What is wrong with the following while statement?

while ( z >= 0 )

sum += z;

Ans:

**The value of the variable z is never changed in the while statement. Therefore, if the loopcontinuation**

**condition ( z >= 0 ) is true, an infinite loop is created. To prevent an infinite loop from**

**occurring, z must be decremented so that it eventually becomes less than 0..**