9

SELF-LEARNING PACKAGE IN

ICT 9

Quarter 2 | Week 4-5

Control Structures

Learning Competency:

Writing codes using control structures in programming.

SSP_TLE-CT8AP-IId-m-3.2

MARICAR R. PORNEL, *Oton NHS*
ZALDY M. TONDO, *Division Science Coordinator*
WRITERS

Ready to Launch!

In the most basic sense, a program is a list of instructions. Control structures are programming blocks that can change the path we take through those instructions.

In module 3, we discuss the different control structures and how it is use in programming. In this lesson, we will continue with our lesson in Control structures, and this time we will write simple programs codes using Java.

Aim at the Target!

At the end of this module you are expected to:

1. Write program code using control structures.

Try This!

Direction. Fill in the blanks. Select the correct answer from the words inside the box.

For	Selection	While
Sequence	Do while	Case

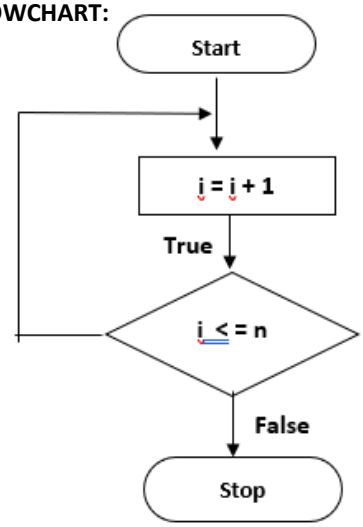
1. The _____ loop evaluations the condition first and then execute the statements.
2. _____ statement is where there is more than one possible choices to choose when trying to solve the problem. Only one process can be carried out.
3. The_____ loop executes the statements first before evaluating the condition.
4. _____allows the computer to decide between two or more different courses of action by testing conditions that occur as the program is running.
5. The _____loop allows a statement to be executed a specified a number of times.

Keep This in Mind!

Direction. Observe carefully the flowchart and the program code and answer the questions in the Analysis section.

Activity 1. Flowchart and program code comparison

FLOWCHART:



Problem: The program will display the numbers from 1 to 100.

PROGRAM CODE:

```
class Main {
  public static void main(String[] args) {
    int i = 1, n = 99;
    while(i <= y) {
      System.out.println(l);
      i++;
    }
  }
}
```

Analysis:

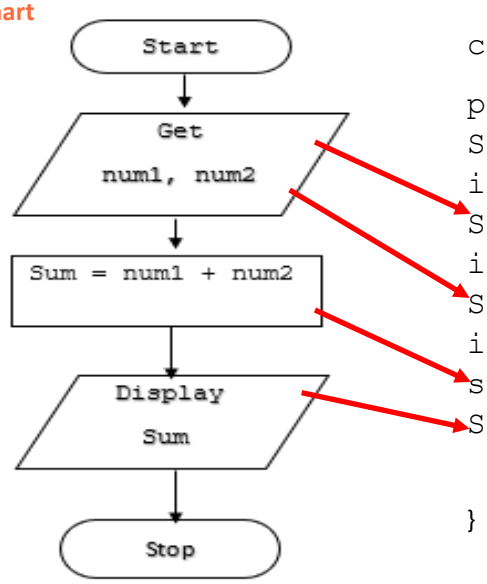
1. Are the flow of instructions in the flowchart the same in the program code? Explain. **Hint.** Review the flowchart of While loop and Do-while loop.
2. Did you find any error with the program code? Explain.(**Hint 1**. Observe what is ask in the problem and compare it with the program code. **Hint 2**. Observe the variable declarations)

Abstraction and Generalization

A. Sequence

Problem: Compute for the sum of two numbers.

Flowchart



Program Code

```
class Input {

public static void main(String[] args) {
Scanner input = new Scanner(System.in);
int sum;
System.out.print("Enter first number:");
int num1 = input.nextInt();
System.out.print("Enter second number:");
int num2= input.nextInt();
sum = (num1 + num2);
System.out.println("the sum is:" + sum);
  }

}
```

B. SELECTION

1. BINARY SELECTION

In binary selection, we use the if ..Else statement.

If .. Else statement tests the condition. It executes the *if block* if condition is true otherwise *else block* is executed.

Syntax:

```
if (condition ) {
   statement
   statement
     ...
 }
  else {
   statement
   statement
     ...
 }
```

True branch

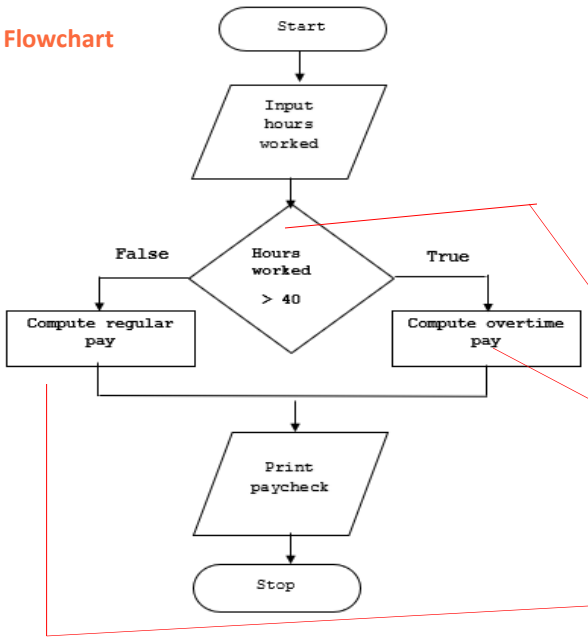
This is executed if the condition

False branch

This is executed if the condition

Problem. Determine whether to compute for the regular pay or overtime pay based on the hours worked of the employee. If hours worked is than greater 40 then compute for overtime, otherwise it will compute for regular pay.

Flowchart



Program Code:

```
 Class Pay
{
  public static void main (String [] args)
  {
  Scanner input = new Scanner(System.in);
System.out.print("Input Hours Worked: "
int hours = input.nextInt();
  if ( hours > 40)
  {
    System.out.println ("Compute Overtime Pay");
      }
         else if (hours < 40)
  {
   System.out.println("Compute Regular Pay");
  }
 }
```

2. Switch Statement

Switch statement tests the value of a variable and compares it with multiple cases. Once the case match is found, a block of statements associated with that particular case is executed.

Each case in a block of a switch has a different name/number which is referred to as an identifier. The value provided by the user is compared with all the cases inside the switch block until the match is found.

If a case match is NOT found, then the default statement is executed, and the control goes out of the switch block.

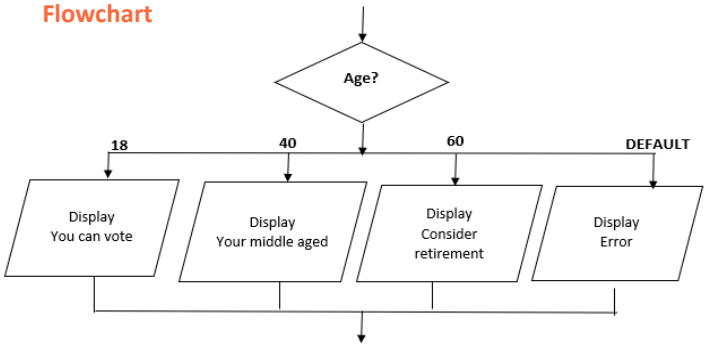
Syntax:

```
     switch( expression )
{
     case value-1:
              Block-1;
              Break;
     case value-2:
              Block-2;
              Break;
     case value-n:
              Block-n;
              Break;
     default:
              Block-1;
              Break;
```

Problem: Write a program that will match the age inputted by the user and will display the corresponding text message.

Age	Text to display
18	You can vote
40	Your middle aged
60	Consider retirement
Age Not in the	Error

Flowchart



Program code

```
 class Age
{
 public static void main(String[]args)
  {
  Scanner in = new Scanner(System.in);
System.out.print("Enter your age:");

int user = in.nextInt();
switch (user) {
     case 18: System.out.println("You can Vote!");
     break;
     case 40:System.out.println("Your Middle Aged!");
     break;
     case 60:System.out.println("Consider Retirement!");
     break;
     default: System.out.println("Error");
   }

 }

}
```

C. Repetitions

1. While loop Statement

The while loop is used to run a specific code until a certain condition is met.

Syntax:

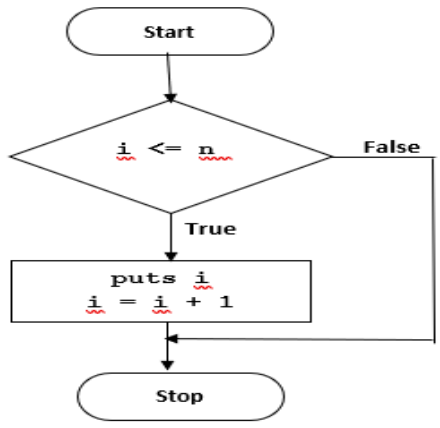
```
while (testExpression)
{
    // body of loop
}
```

How while statement works?

1. A while loop evaluates the **textExpression** inside the parenthesis ().
2. If the **textExpression** evaluates to true, the code inside the while loop is executed.
3. The **textExpression** is evaluated again.
4. This process continues until the **textExpression** is false.
5. When the **textExpression** evaluates to false, the loop stops.

Problem: Write a program that will display numbers from 1 to 5.

Flowchart



Program code

```
class Main {
  public static void main(String[] args) {

    // declare variables
    int i = 1, n = 5;

    // while loop from 1 to 5
    while(i <= n)
 {
      System.out.println(i);
      i++;
    }
  }
}
```

Initialization of the variables

Test expression of while loop

Body of the loop
Print value of `i`

Sequence of while loop
`i++` means to increment the value of `i`

Note. `i++` is a shorthand notation for `i = i + 1`.

OUTPUT:

```
1
2
3
4
5
```

Here, is how the program code of while loop works.

ITERATION	VARIABLE	CONDITION: `i < = n`	ACTION
1st	`i = 1` `n = 5`	true	1 is printed 1 is increased to 2
2nd	`i = 2` `n = 5`	true	2 is printed 1 is increased to 3
3rd	`i = 3` `n = 5`	true	3 is printed 1 is increased to 4
4th	`i = 4` `n = 5`	true	4 is printed 1 is increased to 5
5th	`i = 5` `n = 5`	true	5 is printed 1 is increased to 6
6th	`i = 6` `n = 5`	false	The loop is terminated

2. Do-While loop

The do...while loop is similar to while loop. However, the body of do...while loop is executed once before the test expression is checked.

Syntax:

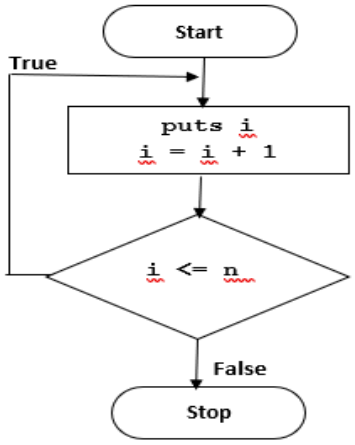
```
do {
      // body of loop
}  while (textExpression)
```

How do-while statement works?

1. The body of the loop is executed at first. Then the **textExpression** is evaluated.
2. If the **textExpression** evaluates to true, the body of the loop inside the do statement is executed again.
3. The **textExpression** is evaluated once again.
4. If the **textExpression** evaluates to true, the body of the loop inside the do statement is executed again.
5. This process continues until the **textExpression** evaluates

Problem: Write a program that will display numbers from 1 to 5.

Flowchart



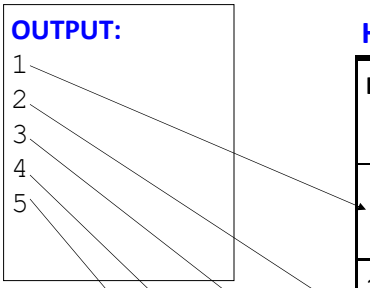
Program code

```
import java.util.Scanner;

class Main {
  public static void main(String[] args) {

    int i = 1, n = 5;

    // do...while loop from 1 to 5
    Do
    {
      System.out.println(i);
      i++;
    } while(i <= n);
  }
}
```



Here, is how the program code of do- while loop works.

ITERATION	VARIABLE	CONDITION: `i < = n`	ACTION
	i = 1 n = 5	Not checked	1 is printed 1 is increased to 2
1st	i = 2 n = 5	true	2 is printed 1 is increased to 3
2nd	i = 3 n = 5	true	3 is printed 1 is increased to 4
3rd	i = 4 n = 5	true	4 is printed 1 is increased to 5
4th	i = 5 n = 5	true	5 is printed 1 is increased to 6
5th	i = 6 n = 5	false	The loop is terminated

3. For Loop

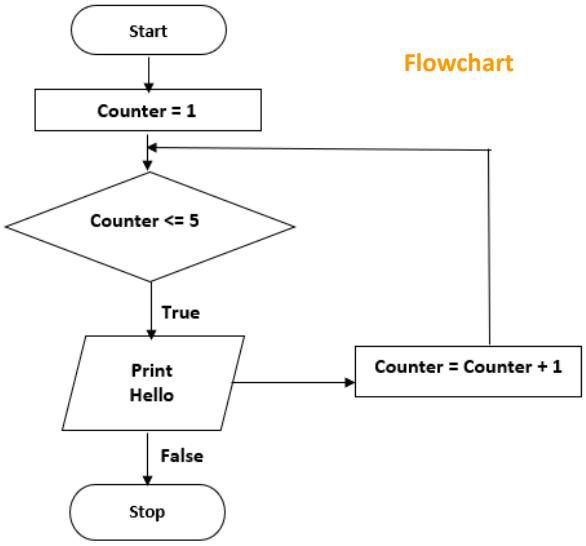
Java for loop is used to run a block of code for a certain number of times.

Syntax:

```
for (initialExpression; testExpression; updateExpression)
{
    // body of the loop
}
```

How For loop statement works?

1. The **initialExpression** initializes and/or declares variables and executes only once.
2. The **condition** is evaluated. If the **condition** is true, the body of the for loop is executed.
3. The **updateExpression** updates the value of **initialExpression**.
4. The **condition** is evaluated again. The process continues until the **condition** is false.



Flowchart

Problem: Write a program that will print a text Hello 5 times.

Program code:

```
class Main {
  public static void main(String[] args) {
    int n = 5;
```

Initialize another variable in a separate statement

Declare and Initialize a loop control variable

Loop continuation condition

Increment

```
  for (int i = 1;   i <= n;    ++i)
      {
      System.out.println("Hello");
    }
  }
}
```

Body of the loop

Note. i++ is a shorthand notation for i = i + 1.

OUTPUT:

Hello
Hello
Hello
Hello
Hello

Here, is how the program code of do- while loop

ITERATION	VARIABLE	CONDITION:	ACTION
1st	i = 1 n = 5	true	Hello is printed i is increased to 2
2nd	i = 2 n = 5	true	Hello is printed i is increased to 3
3rd	i = 3 n = 5	true	Hello is printed i is increased to 4
4th	i = 4 n = 5	true	Hello is printed i is increased to 5
5th	i = 5 n = 5	true	Hello is printed i is increased to 6
6th	i = 6 n = 5	false	The loop is terminated

Application.

Direction. Explain how the program works based on the given program code and output. Copy the format of the table on a separate paper and fill it up with your answers.

Activity 2. Analyzing program code

Problem:

Write a program to display numbers from 6 to 10.

Program code:

```
class Main {
  public static void main(String[] args) {
    int i = 6, n = 10;
    while(i <= n) {
      System.out.println(i);
      i++;
    }
  }
```

OUTPUT

6
7
8
9
10

HINT. See example of while loop

Table

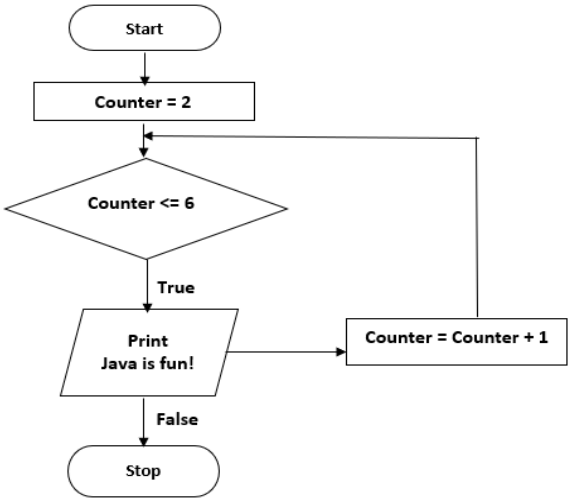
ITERATION	VARIABLE	CONDITION	ACTION

Reinforcement & Enrichment

Direction. Answer the question below.

1. Write the program code of the given flowchart.
2. Write the output of the program code.

Activity 3. Writing program code



HINT. See example of For loop

Reflect

Direction. Complete the sentences below.
After you performed the activity above,

1. I learned that _____.
2. I realized that _____.
3. I hoped that _____.

Assess Your Learning

I. **Multiple Choice**. Select the letter of the correct answer.

1. How many times will the following code print "Welcome to Java"?

```
int x = 10;
for (int i = 5; i <= x; ++i) {
System.out.println("Welcome to Java!");
}
```

a. 4 b. 5 c. 6 d. 10

2. What is the output of the program code?

```
int x = 0;
   while (x < 4) {
   x = x + 1;
  System.out.println("x is " + x);
}
```

A. 1
2
3
4

B.. X is 1
X is 2
X is 3
X is 4

C. x is 1
x is 2
x is 3
x is 4

D. None of The above

5. What is the output of the program code?

```
int i = 51;
   if(i > 50)
System.out.println("Greater than 50");
   else
System.out.println("Less than 50");
System.out.println("Done");
```

A. Greater than 50
B. Less than 50
C. Greater than 50
Done
D. None of the above

II. **Essay (5 points)**

1. How do you compare While loop from Do-while loop?

Program code:

```
switch (course) {
      case 1: System.out.println("Engineering"); break;
      case 2: System.out.println("Nursing"); break;
      case 3: System.out.println("Architecture"); break;
      case 4: System.out.println("Education"); break;
      default: System.out.println("Error");
   }
```

3. What will be the output if the user's input is 8?
   A. Education B. Error C. None of the above
4. Which among the following input number will display the output Architecture?
   A. 1 b. 2 c. 3 d. 4

References & Photo Credits

https://www.homeandlearn.co.uk/java/java_switch_statements.html
https://www.guru99.com/c-switch-case-statement.html#2
https://www.programiz.com/java-programming/do-while-loop
https://www.programiz.com/java-programming/for-loop
https://www.javacodeexamples.com/java-if-else-quiz

Prepared by: MARICAR R. PORNEL, Teacher 1

GLOSSARY:

Operators use in programming

Operator	Meaning	Example
==	Equals	if (x == y)
!=	Not equals	if (x != y)
>	Greater than	if (x > y)
>=	Greater than or equal to	if (x >= y)
<	Less than	if (x < y)
<=	Less than or equal to	if (x <= y)