Experiment No.1
Basic programming constructs like branching and looping
Date of Performance:
Date of Submission:

Aim :- To apply programming constructs of decision making and looping.

Objective :- To apply basic programming constructs like Branching and Looping for solving arithmetic problems like calculating factorial of a no entered by user at command prompt .

Theory :-

Programming constructs are basic building blocks that can be used to control computer programs. Most programs are built out of a fairly standard set of programming constructs. For example, to write a useful program, we need to be able to store values in variables, test these values against a condition, or loop through a set of instructions a certain number of times. Some of the basic program constructs include decision making and looping.

Decision Making in programming is similar to decision making in real life. In programming also, we face some situations where we want a certain block of code to be executed when some condition is fulfilled. A programming language uses control statements to control the flow of execution of a program based on certain conditions. These are used to cause the flow of execution to advance, and branch based on changes to the state of a program.

- if
- if-else
- nested-if
- if-else-if
- switch-case
- break, continue

These statements allow you to control the flow of your program's execution based upon conditions known only during run time.

A loop is a programming structure that repeats a sequence of instructions until a specific condition is met. Programmers use loops to cycle through values, add sums of numbers, repeat functions, and many other things. ... Two of the most common types of loops are the while loop and the for loop. The different ways of looping in programming languages are

- while
- do-while
- for loop
- Some languages have modified for loops for more convenience eg :- Modified for loop in java.

For and while loop is entry-controlled loops. Do-while is an exit-controlled loop.

Code: -

```
import java.io.*;
import java.util.*;

public class Condition
{
   public static void main(String args[])
   {
      Scanner A = new Scanner(System.in);
      System.out.println("Enter 2 numbers:");
      int a = A.nextInt();
      int b = A.nextInt();

      if (a > b)
      {
         System.out.println("The Largest is " + a);
      }
      else
      {
         System.out.println("The largest is " + b);
      }
   }
}
```

Output:

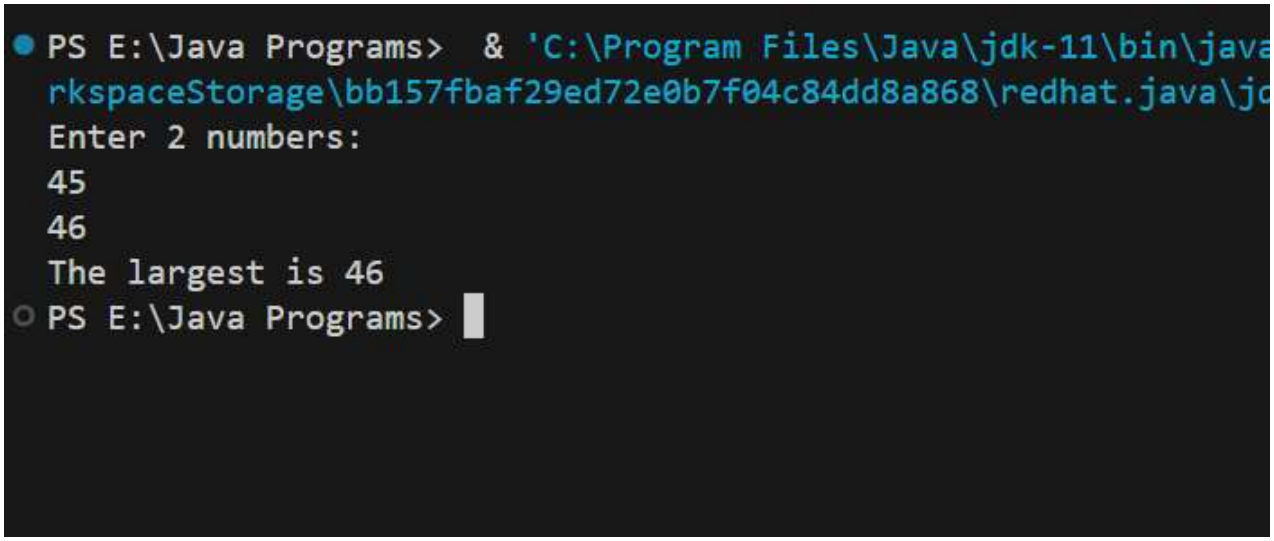
Enter 2 numbers:

45

46

The largest is 46

ScreenShot:



Conclusion:

Comment on how branching and looping useful in solving problem

Ans: Branching allows a program to make decisions based on certain conditions. This is typically done using if, else if, and else statements.

1. Decision Making: Branching enables the program to choose different paths of execution based on conditions. For example, in your code, the if-else statement decides which number is larger.
2. Handling Multiple Scenarios: It allows handling different scenarios and inputs, making the program more flexible and robust.
3. Error Handling: Branching can be used to handle errors and exceptions, ensuring the program can respond appropriately to unexpected situations