KL (DEEMED TO BE UNIVERSITY)

OBJECT ORIENTED PROGRAMMING LABORATORY

21SC1204-R

STUDENT ID: 2100040024 ACADEMIC YEAR: 2023-24

STUDENT NAME:M.H.G.Subhang

Table of Contents

A.Y. 2023-24 LAB/SKILL CONTINUOUS EVALUATION

S.No	Date	Experiment Name	Pre-Lab (10M)	In-Lab (25M)			Post-Lab (10M)	Viva Voce (5M)	Total (50M)	Faculty Signature
				Program/ Procedure (5M)	Data and Results (10M)	Analysis & Inference (10M)				
1.		Introductory Session	-NA-							
2.		Basic Control Structure								
3.		Loop Statements								
4.		Programs On Recursion								
5.		Programs On Arrays								
6.		Constructors								
7.		Method Overloading								
8.		String Buffer Classes								
9.		Inheritance								
10.		Packages								
11.		Exception Handling								
12		Multi-Threading								
13.		Swing package-based event driven programming								

Following is a step by step guide to download and install Eclipse IDE:

Step 1) Installing Eclipse.

Step 2) Click on “Download” button.

Step 3) Click on “Download 64 bit” button.

Step 4) Click on “Download” button.

Step 4) Install Eclipse.

Step 5) Click on Run button.

Step 6) Click on “Eclipse IDE for Java Developers”

Step 7) Click on “INSTALL” button

Step 8) Click on “LAUNCH” button.

Step 9) Click on “Launch” button.

Step 10) Click on “Create a new Java project” link.

Step 11) Create a new Java Project

Write project name.

Click on “Finish button”.

Organization of the STUDENT LAB WORKBOOK

The laboratory framework includes a creative element but shifts the time-intensive aspects outside of the Two-Hour closed laboratory period. Within this structure, each laboratory includes three parts: Prelab, In-lab, and Post-lab.

a. Pre-Lab

The Prelab exercise is a homework assignment that links the lecture with the laboratory period - typically takes 2 hours to complete. The goal is to synthesize the information they learn in lecture with material from their textbook to produce a working piece of software. Prelab Students attending a two-hour closed laboratory are expected to make a good- faith effort to complete the Prelab exercise before coming to the lab. Their work need not be perfect, but their effort must be real (roughly 80 percent correct).

b. In-Lab

The In-lab section takes place during the actual laboratory period. The First hour of the laboratory period can be used to resolve any problems the students might have experienced in completing the Prelab exercises. The intent is to give constructive feedbackso that students leave the lab with working Prelab software - a significant accomplishmenton their part. During the second hour, students complete the In-lab exercise to reinforce the concepts learned in the Prelab. Students leave the lab having received feedback on their Prelab and In-lab work.

c. Post-Lab

The last phase of each laboratory is a homework assignment that is done following the laboratory period. In the Post-lab, students analyze the efficiency or utility of a given system call. Each Post-lab exercise should take roughly 120 minutes to complete.

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE F2ILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

Experiment Title: **Loop Statements**

Aim/Objective: The aim of loop statements is to allow programmers to repeat a block of code multiple times. Loop statements provide a way to execute a set of instructions iteratively until a specific condition is met.

Description:

Loop statements execute a set of instructions until a specific condition is met or until a certain number of iterations are completed. Here is a description of the three common loop statements

Pre-Requisites:

1. Java Development Kit (JDK)
2. Text Editor
3. Integrated Development Environment (IDE)

Pre-Lab:

1) Explain Switch Case With Syntax?

Ans:

Switch Case is a control flow statement that allows you to choose between multiple code blocks based on the value of an expression. The syntax is as follows:

```
switch (expression) {
  case value1:
    // Code block executed when expression matches value1
    break;
  case value2:
    // Code block executed when expression matches value2
    break;
  // Add more cases as needed
  default:
    // Code block executed if none of the cases match
}
```

2) Explain jump and break?

Ans:

Jump and Break:

"Jump" refers to changing the flow of program execution to another section, usually achieved with keywords like goto (in some languages). However, excessive use of jumps is discouraged as it can make code hard to read and maintain.

"Break" is used to exit a loop or switch case prematurely. When encountered, the loop stops iterating or the switch case exits immediately, continuing with the code after the loop or switch.

3) What are pre post exist loop?

Ans:

Pre, Post, and Exit Loops: I'm not familiar with the term "pre, post, and exit loops." It's possible that you meant "pre-test," "post-test," and "exit-controlled" loops, which are loop control structures. Let me explain them briefly:

Pre-Test Loop (while loop):

```
while (condition) {
  // Code block executed while the condition is true
}
```

Post-Test Loop (do-while loop):

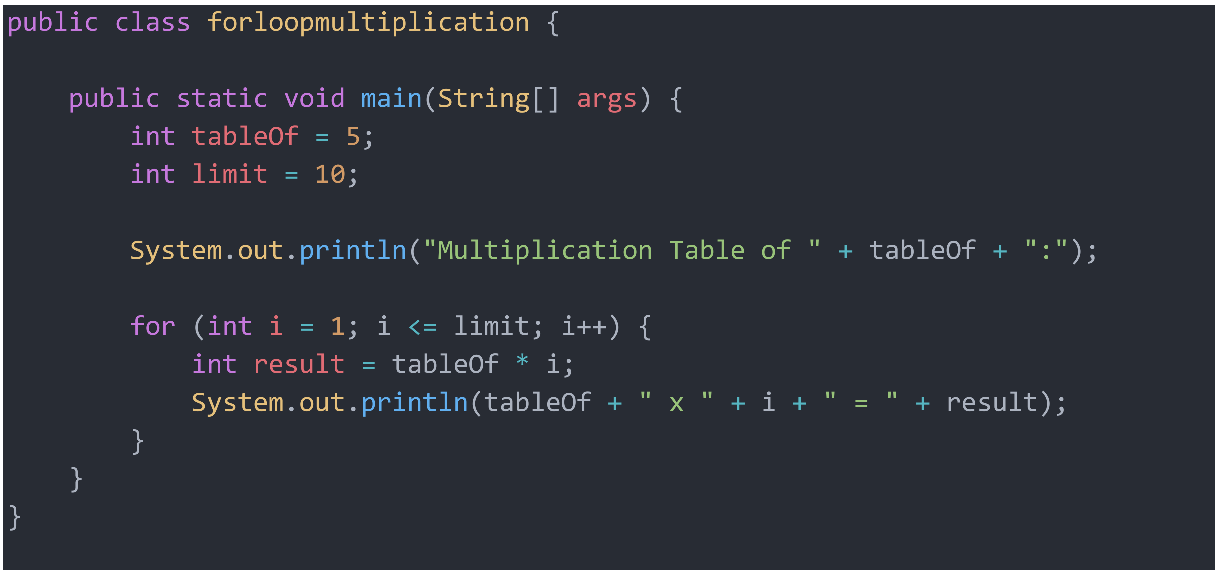
```
do {
  // Code block executed while the condition is true
} while (condition);
```

In-Lab:
1) To solve the Loop Statements

.Java Program to demonstrate the example of Switch statement .where we are printing month name for the given number
. Java program to generate a multiplication table using for- loop

Procedure/Program:

```
import java.util.Scanner;

public class monthselectswitch {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the month number (1-12): ");
        int monthNumber = scanner.nextInt();

        switch (monthNumber) {
            case 1:
                System.out.println("January");
                break;
            case 2:
                System.out.println("February");
                break;
            case 3:
                System.out.println("March");
                break;
            case 4:
                System.out.println("April");
                break;
            case 5:
                System.out.println("May");
                break;
            case 6:
                System.out.println("June");
                break;
            case 7:
                System.out.println("July");
                break;
            case 8:
                System.out.println("August");
                break;
            case 9:
                System.out.println("September");
                break;
            case 10:
                System.out.println("October");
                break;
            case 11:
                System.out.println("November");
                break;
            case 12:
                System.out.println("December");
                break;
            default:
                System.out.println("Invalid month number. Please enter a
number between 1 and 12.");
        }
        scanner.close();
    }
}
```

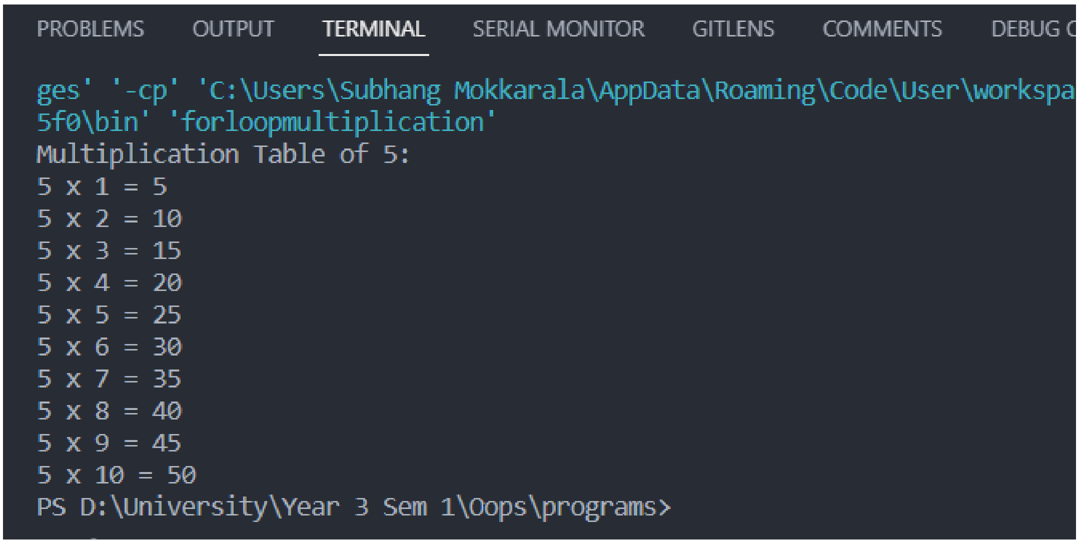


```java
public class forloopmultiplication {

    public static void main(String[] args) {
        int tableOf = 5;
        int limit = 10;

        System.out.println("Multiplication Table of " + tableOf + ":");

        for (int i = 1; i <= limit; i++) {
            int result = tableOf * i;
            System.out.println(tableOf + " x " + i + " = " + result);
        }
    }
}
```

- **Data and Results:**

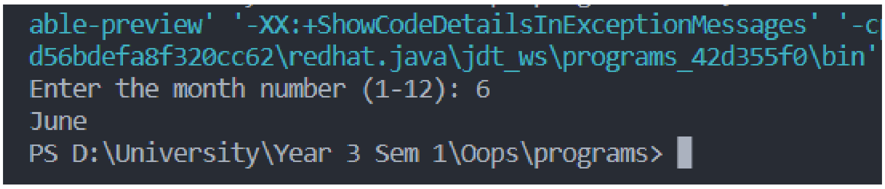
19-07-2023

1.



```
ges' '-cp' 'C:\Users\Subhang Mokkarala\AppData\Roaming\Code\User\workspa
5f0\bin' 'forloopmultiplication'
Multiplication Table of 5:
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
PS D:\University\Year 3 Sem 1\Oops\programs>
```

2.



```
able-preview' '-XX:+ShowCodeDetailsInExceptionMessages' '-c
d56bdefa8f320cc62\redhat.java\jdt_ws\programs_42d355f0\bin'
Enter the month number (1-12): 6
June
PS D:\University\Year 3 Sem 1\Oops\programs>
```

- **Analysis and Inferences:**
- **Analysis and Inferences:**

The given Java program uses a for-loop to generate a multiplication table for a user-inputted number. It first prompts the user to enter a number using the Scanner class and stores it in the variable "number." The program then proceeds to print the multiplication table for this number from 1 to 10 using a for-loop. During each iteration of the loop, it calculates the product of the number and the loop index "i," printing the multiplication expression and the result. The program provides a simple and efficient way to generate and display the multiplication table for any given number, helping users understand the basic concept of multiplication and its applications. By entering different numbers, users can quickly generate multiplication tables tailored to their specific needs. Overall, this program is a helpful educational tool and showcases the use of loops and user input in Java programming.

Sample VIVA-VOCE Questions (In-Lab):

1. **Write a logic to find the factorial of a given number using for loops.**

To find the factorial of a given number using a for-loop, you can follow this logic:
Initialize a variable factorial to store the result and set it to 1 initially.
Take the input number for which you want to find the factorial.
Run a for-loop from 1 to the input number.
In each iteration, multiply the current value of factorial with the loop index.
Update the value of factorial with the new result in each iteration.
After the loop completes, the value of factorial will be the factorial of the given number.

2. How many times 'Hello' is printed?

```
public class CppBuzz {
public static void main(String[] args){
 for(int i = 0; i<5; i++)
 {
 System.out.println("Hello");
 i+=2;
 }
}
}
```

Out put:

```
Hello
Hello
Hello
```

3. Is their any difference in while and do while?

while loop:

The while loop is a pre-test loop, meaning that the loop condition is checked before each iteration. If the condition is true, the loop body is executed, and if the condition is false initially, the loop body is never executed.

do-while loop:

The do-while loop is a post-test loop, meaning that the loop body is executed at least once before the loop condition is checked. After executing the loop body, the loop condition is evaluated. If the condition is true, the loop continues, and if it is false, the loop terminates.

4. Difference between break and continue?

break: Terminates the loop immediately when a condition is met.
continue: Skips the rest of the current iteration and moves to the next iteration when a condition is met.

5. What is pre-checking Loop

A pre-checking loop checks the loop condition before executing the loop body. If the condition is false initially, the loop body is not executed. It is often implemented using the while loop.

Post-Lab:

. Java program to illustrate the use of switch statements with strings

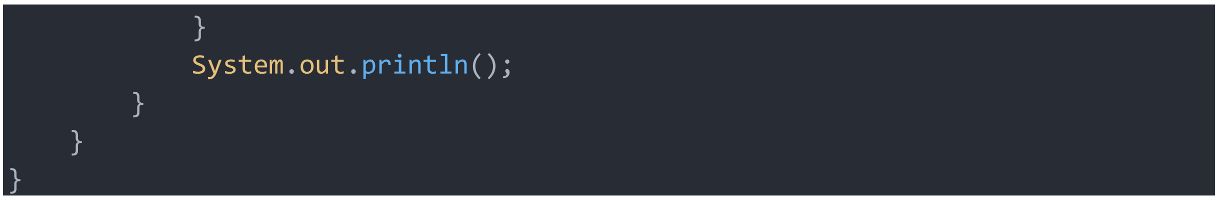
. Java Program to Print Hollow Square Star Pattern

- **Procedure/Program:**

1.)
```
public class StringSwitchExample {

    public static void main(String[] args) {
        String dayOfWeek = "Monday";
        switch (dayOfWeek) {
            case "Monday":
                System.out.println("It's the start of the week!");
                break;
            case "Tuesday":
            case "Wednesday":
            case "Thursday":
                System.out.println("It's a workday.");
                break;
            case "Friday":
                System.out.println("TGIF! It's Friday!");
                break;
            case "Saturday":
            case "Sunday":
                System.out.println("It's the weekend. Time to relax!");
                break;
            default:
                System.out.println("Invalid day of the week.");
        }
    }
}
```

2.

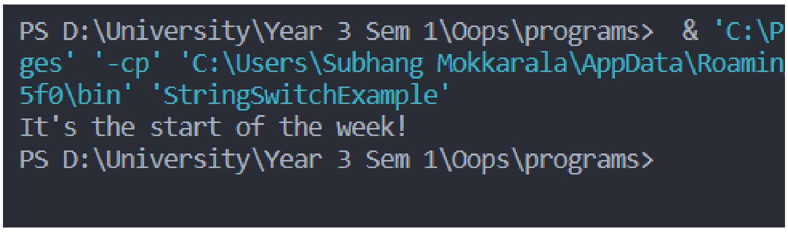
```
public class HollowSquarePattern {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the number of rows: ");
        int rows = scanner.nextInt();

        for (int i = 1; i <= rows; i++) {
            for (int j = 1; j <= rows; j++) {
                if (i == 1 || i == rows || j == 1 || j == rows) {
                    System.out.print("* ");
                } else {
                    System.out.print("  ");
                }
```



```
            }
            System.out.println();
        }
    }
}
```

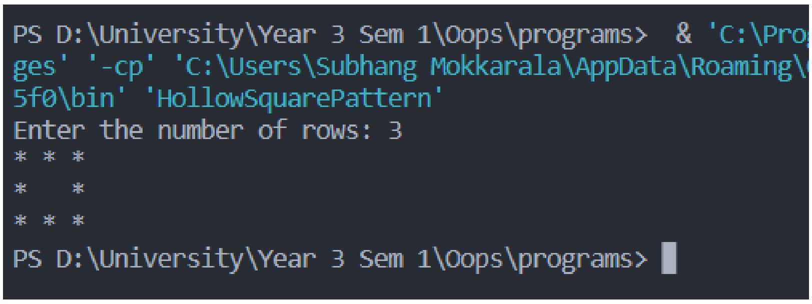
- **Data and Results:**

26-07-2023
1.)



```
PS D:\University\Year 3 Sem 1\Oops\programs>  & 'C:\P
ges' '-cp' 'C:\Users\Subhang Mokkarala\AppData\Roamin
5f0\bin' 'StringSwitchExample'
It's the start of the week!
PS D:\University\Year 3 Sem 1\Oops\programs>
```

2.)



```
PS D:\University\Year 3 Sem 1\Oops\programs>  & 'C:\Pro
ges' '-cp' 'C:\Users\Subhang Mokkarala\AppData\Roaming\
5f0\bin' 'HollowSquarePattern'
Enter the number of rows: 3
* * *
*   *
* * *
PS D:\University\Year 3 Sem 1\Oops\programs> █
```

- **Analysis and Inferences:**

- Loop statements in programming enable the repetition of a code block until a certain condition is satisfied. They save coding effort, enhance efficiency, and are crucial for automating repetitive tasks.

Experiment #	<TO BE FILLED BY STUDENT>	Student ID	<TO BE F2ILLED BY STUDENT>
Date	<TO BE FILLED BY STUDENT>	Student Name	<TO BE FILLED BY STUDENT>

Evaluator Remark (if Any):	**Marks Secured: _____out of 50**
	Signature of the Evaluator with Date

Evaluator MUST ask Viva-voce prior to signing and posting marks for each experiment.