IARE

IARE
INSTITUTE OF AERONAUTICAL ENGINEERING
(An Autonomous Institute affiliated to JNTUH, Hyderabad)
Dundigal, Hyderabad - 500 043

LABORATORY WORK BOOK

Name of the Student : RACHERLA SANTHOSH

Roll Number										
2	3	9	5	1	A	1	2	6	3

Class : IT-B Semester : 03

Course Code : ACSD11 Course Name : DS Laboratory

Name of the Course Faculty : Ms. K. Laxminarayanamma Faculty ID : IARE10033

Exercise Number : 01 Week Number : 01 Date : 27/08/2024

S. No.	Exercise Number	EXERCISE NAME	MARKS AWARDED					
			Aim/ Preparation	Algorithm / Procedure / Performance in the Lab	Source Code / Calculations and Graphs	Program Execution / Results and Error Analysis	Viva-Voce	Total
			4	4	4	4	4	20
1	1.1	Sum of last Digits	4					
2	1.2	Is N an exact multiple of M.						
3	1.3	Combine Strings						
4	1.4	Even Or Odd						
5	1.5	Second Last Digit						
6	1.6	Alternate String Combiner						
7	1.7	Padovan Sequence						
8	1.8	Leaders In An Array						
9	1.9	Find Value Of A Number Raised to its Reverse.						
10	1.10	Mean of Array using Recursion.						
11			4	4	4	4	4	20
12								

Santhosh
Signature of the Student

Laxmi
Signature of the Faculty

1. Getting Started Exercises.

1.1 Sum Of Last Digits Of Two Given Numbers

AIM :- Write the Program to add the last digits of two given numbers.

PROGRAM :-

```
import java.util.Scanner;
Public class AddLastDigits {
    Public Static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter the first number:");
        int a = scanner.nextInt();
        System.out.print("Enter the second number:");
        int b = scanner.nextInt();
        int lastDigit1 = Math.abs(a % 10);
        int lastDigit2 = Math.abs(b % 10);
        int result = lastDigit1 + lastDigit2;
```

```
        System.out.println("Sum of last digits : " +
                                    result);
        Scanner.close();
    }
}
```

RESULT :-

INPUT : 267 OUTPUT : 11

154

1.2 Is N an exact multiple Of M :-

AIM :- Write a function that accepts two Parameters and finds whether the first Parameter is an exact multiple of the Second Parameter. If the first parameter is an exact multiple of the Second Parameter, the function should return 2 else it should return 1. If either of

the parameters are 0, the function should return 3.

PROGRAM :-

```
import java.util.Scanner;
Public class ExactMultiple {
    Public static void main (String[] args) {
        Scanner scanner = new Scanner (System.in);
        System.out.print ("Enter the first number: ");
        int a = scanner.nextInt();
        System.out.print ("Enter the second number: ");
        int b = scanner.nextInt();
        int result;
        if (a==0 || b==0) {
            result = 3;
        } else if (a%b==0) {
            result = 2;
```

```
} else {
    result = 1;
}
System.out.println(result);
scanner.close();
}
}
```

RESULT :-

INPUT : num1 = 10, num2 = 5

OUTPUT : 2.

1.3 Combine Strings :-

AIM :- Write a Program on given two Strings a and b, return a new String of the form short + long + short, with the shorter String on the outside and the longer String in the inside. The

Strings will not be the same length, but they may be empty (length 0).

PROGRAM :-

```
import java.util.Scanner;
Public class StringCombiner {
  Public static void main (String[] args) {
    Scanner scanner = new Scanner (System.in);
    System.out.print ("Enter the first string:");
    String a = scanner.nextLine();
    System.out.print ("Enter the second string:");
    String b = scanner.nextLine();
    String result;
    if (a.length() <= b.length()) {
      result = a + b + a;
    } else {
      result = b + a + b;
```

```
    }
    System.out.println("Result:" + result);
    scanner.close();
  }
}
```

RESULT :-

INPUT : Enter the first String : "iare"
Enter the second String : "college"

OUTPUT : "iare college iare"

1.4 Even Or Odd :-

AIM :- Write a function that accepts 6 input Parameters. The first 5 input Parameters are of type int. The sixth input parameter is of type String. If the Sixth parameter contains the value "even", the function is supposed to return the count of how many

of the first five input parameters are even. If the sixth parameter contains the value "Odd", the function is supposed to return the count of how many of the first five input parameters are Odd.

PROGRAM :-

```
import java.util.Scanner;
Public class CountEvenOdd {
  Public Static void main (String[] args) {
    Scanner scanner = new Scanner (System.in);
    System.out.print ("num1 : ");
    int num1 = scanner.nextInt();
    System.out.print ("num2 : ");
    int num2 = scanner.nextInt();
    System.out.print ("num3 : ");
    int num3 = scanner.nextInt();
    System.out.print ("num4 : ");
```

```
int num4 = scanner.nextInt();
System.out.print("num 5: ");
int num5 = scanner.nextInt();
System.out.print("type: ");
String type = scanner.next();
int count = 0;
if (type.equalsIgnoreCase("even")) {
    if (num1 % 2 == 0) count++;
    if (num2 % 2 == 0) count++;
    if (num3 % 2 == 0) count++;
    if (num4 % 2 == 0) count++;
    if (num5 % 2 == 0) count++;
} else if (type.equalsIgnoreCase("odd")) {
    if (num1 % 2 != 0) count++;
    if (num2 % 2 != 0) count++;
    if (num3 % 2 != 0) count++;
    if (num4 % 2 != 0) count++;
```

```
        if (num5 % 2 != 0) count++;
    } else {
        System.out.println("please enter 'even' or
                                'odd'");
    }
    System.out.println("Count: " + count);
    Scanner.close();
  }
}
```

RESULT :-

INPUT : num1 = 12;
num2 = 17;
num3 = 19;
num4 = 14;
num5 = 115;
type = "odd"

OUTPUT : 3

ROLL NUMBER :

1.5 Second Last Digit Of A Given Number :-

AIM :- Write a function that returns the second last digit of the given number. 'Second' last digit is being referred to the digit in the tens place in the given number.

PROGRAM :-

```
import java.util.Scanner;
class SecondLastDigit {
    Public Static void main (String[] args) {
        Scanner scanner = new Scanner (System.in);
        System.out.print ("Enter a number: ");
        int num = scanner.nextInt();
        num = Math.abs (num);
        int SecondLastDigit;
        if (num < 10) {
            SecondLastDigit = -1;
```

```
    } else {
        secondLastDigit = (num / 10) % 10;
    }
    System.out.println("The second last digit is: "
                        + secondLastDigit);
    scanner.close();
  }
}
```

RESULT :-

① INPUT : 197

OUTPUT : 9

② INPUT : 5

OUTPUT : -1

③ INPUT : -197

OUTPUT : 9

1.6 Alternate String Combiner :-

AIM :- Write a Program on two Strings a and b, Print a new String which is made of the following combination - first character of a, the first character of b, second character of a, second character of b and so on. Any characters left, will go to the end of the result.

PROGRAM :-

```
import java.util.Scanner;
Public class MergeStrings {
    Public Static void main (String[] args) {
        Scanner scanner = new Scanner (System.in);
        System.out.print ("Enter the first String :");
        String a = scanner.nextLine();
        System.out.print ("Enter the Second String :");
```

```
String b = scanner.nextLine();
String result = "";
int len1 = a.length();
int len2 = b.length();
int minLength = Math.min(len1, len2);
for (int i=0; i< minLength; i++) {
    result += a.charAt(i);
    result += b.charAt(i);
}
if (len1 > len2) {
    result += a.substring(minLength);
} else if (len2 > len1) {
    result += b.substring(minLength);
}
System.out.println("Merged String:" +result);
scanner.close();
}
}
```

RESULT :-

INPUT : "Hello, World"

OUTPUT : "HWeolrllod".

1.7 Padovan Sequence :-

AIM :- Write a Program on the Padovan Sequence, where each term is the Sum of the two Proceding terms, Similar to the Fibonacci Sequence. However, the Padovan Sequence has different initial conditions and exhibits different growth Patterns. the first few terms of the Padovan Sequence are : 1, 1, 1, 2, 2, 3, 4, 5, 7, 9, 12, ...

PROGRAM :-

```
Public class PadovanSequence {
    Public Static void main (String[] args) {
        int numTerms = 10;
```

```
int [] padovan = new int [numTerms] ;
if (numTerms > 0) Padovan [0] = 1 ;
if (numTerms > 1) Padovan [1] = 1 ;
if (numTerms > 2) Padovan [2] = 1 ;
for (int i=3 ; i < numTerms ; i++) {
    Padovan [i] = padovan [i-2] + padovan [i-3] ;
}
System . out . println (" Padovan sequence :") ;
for (int i=0 ; i < numTerms ; i++) {
    System . out . print ( Padovan [i] + " ") ;
}
}
}
```

RESULT :-

INPUT : num = 10.

OUTPUT : Padovan Sequence up to 10 :

1 1 1 2 2 3 4 0 5 7 9 12 .

1.8 Leaders In An Array :-

AIM :- Write a program on an array arr of n positive integer. Your task is to find all the leaders in the array. An element of the array is considered a leader if it is greater than all the elements on its right side or if it is equal to the maximum element on its right side. The rightmost element is always a leader.

PROGRAM :-

```
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;
Public class LeadersInArray {
    Public Static void main (String[] args) {
        int[] arr = {16, 17, 4, 3, 5, 2};
        List<Integer> leaders = findLeaders(arr);
        System.out.println("Leaders in the array:");
        for (int leader : leaders) {
            System.out.print(leader + " ");
        }
}
```

```
Public static List <Integer> findLeaders(int[] arr) {
    List <Integer> leaders = new ArrayList <>();
    if (arr == null || arr.length == 0) {
        return leaders;
    }
    int n = arr.length;
    int maxFromRight = arr[n-1];
    leaders.add(maxFromRight);
    for (int i = n-2; i >= 0; i--) {
        if (arr[i] >= maxFromRight) {
            leaders.add(arr[i]);
            maxFromRight = arr[i];
        }
    }
    Collections.reverse(leaders);
    return leaders;
    }
}
```

RESULT :-

INPUT : n=6, arr[] = (16, 17, 4, 3, 5, 2) OUTPUT : 17 5 2

1.9 Find the Value Of A Number Raised To its Reverse :

AIM :- Write a Program on a number N & its reverse R. The task is to find the number obtained when the number is raised to the power of its own reverse.

PROGRAM :-

```
import java.math.BigInteger;
Public class PowerOfReverse {
    Public static void main(String[] args) {
        int number = 12;
        int reverse = reverseNumber(number);
        System.out.println("Number :" + number);
        System.out.println("Reverse :" + reverse);
        System.out.println("Result of" + number + "raised to the power of"
                + reverse + "is :" + result);
    }
    Public static ~~void main~~ int reverseNumber(int num) {
        int reversed = 0;
        while (num != 0) {
            int digit = num % 10;
            reversed = reversed * 10 + digit;
            num /= 10;
        }
```

```
        return result reversed;
    }
}
```

RESULT :- INPUT : N = 2, R = 2 OUTPUT : 4

1.10 Mean Of Array Using Recursion :-

AIM :- Find the Mean Of the elements of the Array.

PROGRAM :-

```
Public class MeanOf Array {
    Public static void main (String [] args) {
        int [] arr = {10, 20, 30, 40, 50};
        double mean = calculate Mean (arr);
        System.out.println ("Mean of the array :" + mean);
    }
    Public static double calculate Mean (int [] arr) {
        int sum = 0;
        for (int num : arr) {
            sum += num;
        }
        return (double) sum / arr.length; } }
```

RESULT :- INPUT : 1 2 3 OUTPUT : 2.0

VIVA VOCE :-

1. What is a Variable in Java ?

A) A Variable in Java is a container that holds data values. It has a data type that determines the kind of value it can store, such as 'int' for integers or 'String' for text.

2. What is an Array in Java ?

A) An Array is a collection of elements, all of the same type, stored in a contiguous memory location.

Ex: `int[] numbers = {1, 2, 3, 4, 5};`

3. What is an 'if' statement ?

A) An 'if' statement is a conditional statement that executes a block of code if a specified condition is true :

```
if (x > 10) {
    System.out.println("x is greater than 10");
}
```

4. What is a loop in Java ?

A) A loop is a control structure that repeatedly executes a block of code as long as a specified condition is True.

Ex: 'for', 'while', 'Do - while' loops.