**ASSIGNMENT 1**

**Set A**

1. **Write a java program to accept a number from command prompt and display factorial of a number Accept number using BufferedReader class.**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.io.IOException;

public class factorial {

public static void main(String[] args) throws IOException{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in)); System.out.println("Enter a number:");

int num = Integer.parseInt(br.readLine());

int fact = 1;

for (int i = 1; i <= num; i++) {

fact \*= i;

}

System.out.println("The factorial of " + num + " is: " + fact);

}

}

Output: Enter a number: 5 The factorial of 5 is: 120

**2) Write a Java program to accept a decimal number and convert it to binary number and vice-versa.(use BufferedReader class).**

import java.io.BufferedReader;

import java.io.InputStreamReader;

import java.io.IOException;

public class SimpleConverter {

public static void main(String[] args) throws IOException

{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

System.out.print("Enter a decimal number: ");

String decimalInput = br.readLine();

int decimal = Integer.parseInt(decimalInput);

String binary = Integer.toBinaryString(decimal);

System.out.println("Binary equivalent: " + binary);

System.out.print("Enter a binary number: ");

String binaryInput = br.readLine(); as a string

int decimalValue = Integer.parseInt(binaryInput, 2);

System.out.println("Decimal equivalent: " + decimalValue);

}

}

Output:

Enter a decimal number: 5

Binary equivalent: 101

Enter a binary number: 1001

Decimal equivalent: 9

=== Code Execution Successful ===

**3) Write a java program to find the maximum and minimum value of an array.**

public class MinMax {

public static void main(String[] args) {

int[] arr = {50, 20, 90, 10, 70};

int max = arr[0];

int min = arr[0];

for (int i = 1; i < arr.length; i++) {

if (arr[i] > max) {

max =arr[i];

}

if (arr[i] < min) {

min = arr[i];

}

}

System.out.println("Maximum value: " + max);

System.out.println("Minimum value: " + min);

}

}

Output:

Maximum value: 90

Minimum value: 10

=== Code Execution Successful ===

**4) Find GCD (Greatest Common Divisor) and LCM (Least Common Multiple) of Two Numbers) Accept input through command line argument).**

public class GCDLCM {

public static void main(String[] args) {

if (args.length != 2) {

System.out.println("Please provide two integers as command-line arguments.");

return;

}

int num1 = Integer.parseInt(args[0]);

int num2 = Integer.parseInt(args[1]);

int gcd = num1;

int temp = num2;

while (temp != 0) {

int remainder = gcd % temp;

gcd = temp;

temp = remainder;

}

int lcm = (num1 \* num2) / gcd;

System.out.println("GCD="+ gcd);

System.out.println("LCM="+ lcm);

}

}

Output:

Please provide exactly two integers as command-line arguments.

=== Code Execution Successful ===

**SET C**

1. **Write a Java program to find common elements between two integer arrays.**

public class CommonElements {

public static void main(String[] args) {

int[] arr1 = {1, 2, 3, 4, 5};

int[] arr2 = {3, 5, 7, 9};

System.out.println("Common elements are:");

for (int i = 0; i < arr1.length; i++) {

for (int j = 0; j < arr2.length; j++) {

if (arr1[i] == arr2[j]) {

System.out.println(arr1[i]);

}

}

}

}

}

Output:

Common elements are:

3

5

=== Code Execution Successful ===

1. **Write a java program to Merge Two Sorted Arrays**

public class MergeSortedArrays {

public static void main(String[] args) {

// Example arrays (already sorted)

int[] arr1 = {1, 3, 5, 7, 9};

int[] arr2 = {2, 4, 6, 8, 10};

// Create a new array to store the merged result

int[] mergedArray = new int[arr1.length + arr2.length];

// Variables to keep track of positions in arr1, arr2, and mergedArray

int i = 0, j = 0, k = 0;

// Merge the arrays into mergedArray

while (i < arr1.length && j < arr2.length) {

if (arr1[i] < arr2[j]) {

mergedArray[k] = arr1[i];

i++;

} else {

mergedArray[k] = arr2[j];

j++;

}

k++;

}

// Copy remaining elements from arr1 if any

while (i < arr1.length) {

mergedArray[k] = arr1[i];

i++;

k++;

}

// Copy remaining elements from arr2 if any

while (j < arr2.length) {

mergedArray[k] = arr2[j];

j++;

k++;

}

// Print the merged array

for (int num : mergedArray) {

System.out.print(num + " ");

}

}

}

Output:

1 2 3 4 5 6 7 8 9 10

=== Code Execution Successful ===

1. **Write a java program to design a calculator using a switch case.**

import java.util.Scanner;

public class Calculator {

public static void main(String[] args) {

// Create a Scanner object to take input from the user

Scanner scanner = new Scanner(System.in);

// Ask for user input: two numbers and the operation

System.out.print("Enter first number: ");

double num1 = scanner.nextDouble();

System.out.print("Enter second number: ");

double num2 = scanner.nextDouble();

System.out.println("Choose an operation:");

System.out.println("1. Addition (+)");

System.out.println("2. Subtraction (-)");

System.out.println("3. Multiplication (\*)");

System.out.println("4. Division (/)");

int choice = scanner.nextInt();

// Variable to store the result

double result;

// Switch case to perform the selected operation

switch (choice) {

case 1: // Addition

result = num1 + num2;

System.out.println("The result of addition is: " + result);

break;

case 2: // Subtraction

result = num1 - num2;

System.out.println("The result of subtraction is: " + result);

break;

case 3: // Multiplication

result = num1 \* num2;

System.out.println("The result of multiplication is: " + result);

break;

case 4: // Division

// Check for division by zero

if (num2 != 0) {

result = num1 / num2;

System.out.println("The result of division is: " + result);

} else {

System.out.println("Error: Division by zero is not allowed.");

}

break;

default:

System.out.println("Invalid choice. Please choose a valid operation.");

break;

}

// Close the scanner

scanner.close();

}

}

Output:

Enter first number: 4

Enter second number: 5

Choose an operation:

1. Addition (+)

2. Subtraction (-)

3. Multiplication (\*)

4. Division (/)

3

The result of multiplication is: 20.0

=== Code Execution Successful ===

1. **Write a java program to find prime no and add it into the array.**

public class PrimeNumbers {

public static void main(String[] args) {

int limit = 50; // Define the range up to which we want to find prime numbers

int[] primes = new int[limit]; // Array to store prime numbers

int index = 0; // To keep track of how many primes we have found

// Loop through numbers from 2 to 'limit'

for (int num = 2; num <= limit; num++) {

boolean isPrime = true; // Assume the number is prime

// Check if the number is divisible by any number between 2 and num-1

for (int i = 2; i < num; i++) {

if (num % i == 0) {

isPrime = false; // If divisible, it's not prime

break;

}

}

// If the number is prime, add it to the array

if (isPrime) {

primes[index] = num;

index++; // Move to the next position in the array

}

}

// Print the prime numbers found

System.out.print("Prime numbers: ");

for (int i = 0; i < index; i++) {

System.out.print(primes[i] + " ");

}

}

}

Output:

Prime numbers: 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47

=== Code Execution Successful ===