|  |  |
| --- | --- |
| **Program 1.1** | **Date: 01/01/2024** |
| **Write a Java program to check whether a string is palindrome or not.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.\*;  public class P7\_1\_palindrome {  public static void main(String args[]) {  Scanner s = new Scanner(System.in);  int i;  boolean flag = true;  try {  System.out.println("\nEnter a string to check for palindrome : ");  String str = s.nextLine();  str = str.toLowerCase(); // coverts entered string to lowercase  // Comparing one character at a time till middle of the string is reached  for (i = 0; i < (str.length() / 2); i++) {  if (str.charAt(i) != str.charAt(str.length() - i - 1)) {  flag = false;  break;  }  }  if (flag) {  System.out.println("\nstring " + str + " is Palindrome.");  } else {  System.out.println("\nstring " + str + " is Not Palindrome.");  }  } catch (Exception e) {  System.out.println("\nError : " + e);  }  }  } |

**OUTPUT :**

|  |
| --- |
| E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)\JAVA LAB>java P7\_1\_palindrome  Enter a string to check for palindrome :  malayalam  string malayalam is Palindrome.  E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)\JAVA LAB> |

|  |  |
| --- | --- |
| **Program 1.2** | **Date: 02/01/2024** |
| **Write a Java program to multiply two matrices.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.\*;  public class P7\_2\_matrix\_multiply {  public static void main(String args[]) {  Scanner s = new Scanner(System.in);  int row1, col1, row2, col2;  int i, j;  try {  // MATRIX 1  System.out.println("\nEnter the number of rows of First matrix :");  row1 = s.nextInt();  System.out.println("\nEnter the number of columns of First matrix :");  col1 = s.nextInt();  System.out.println("\nEnter the Elements of First matrix : ");  int[][] matrix1 = new int[row1][col1];  for (i = 0; i < row1; i++) {  for (j = 0; j < col1; j++) {  matrix1[i][j] = s.nextInt();  }  }  // MATRIX 2  System.out.println("\nEnter the number of rows of Second matrix :");  row2 = s.nextInt();  System.out.println("\nEnter the number of columns of Second matrix :");  col2 = s.nextInt();  System.out.println("\nEnter the Elements of Second matrix : ");  int[][] matrix2 = new int[row2][col2];  for (i = 0; i < row2; i++) {  for (j = 0; j < col2; j++) {  matrix2[i][j] = s.nextInt();  }  }  // no.of columns of First matrix and no.of rows of Second matrix should be same  // for matrix multiplication posssible.  if (col1 != row2) {  System.out.println("\nMatrix multiplication is not possible !");  } else {  // RESULT MATRIX  int[][] result\_matrix = multiply\_matrix(matrix1, matrix2);  System.out.println("\nResult Matrix is --->");  for (i = 0; i < row1; i++) {  for (j = 0; j < col2; j++) {  System.out.print(result\_matrix[i][j] + "\t");  }  System.out.println("\n");  }  }  } catch (Exception e) {  System.out.println("\nError : " + e);  } finally {  s.close();  }  }  public static int[][] multiply\_matrix(int[][] matrix1, int[][] matrix2) {  int row1, col1, col2;  int i, j, k;  row1 = matrix1.length;  col2 = matrix2[0].length;  col1 = matrix1[0].length;  int[][] result = new int[row1][col2];  // multiplying  for (i = 0; i < row1; i++) {  for (j = 0; j < col2; j++) {  for (k = 0; k < col1; k++) {  result[i][j] += matrix1[i][k] \* matrix2[k][j];  }  }  }  return result;  }  } |

**OUTPUT :**

|  |
| --- |
| E:\MUHAMMAD ANSHAD P A\JAVA\JAVA LAB>java P7\_2\_matrix\_multiply  Enter the number of rows of First matrix :  2  Enter the number of columns of First matrix :  2  Enter the Elements of First matrix :  1  2  3  4  Enter the number of rows of Second matrix :  2  Enter the number of columns of Second matrix :  2  Enter the Elements of Second matrix :  4  5  6  7  Result Matrix is --->  16 19  36 43 |

|  |  |
| --- | --- |
| **Program 1.3** | **Date: 04/01/2024** |
| **Write a Java program to find the transpose of a matrix.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.Scanner;  public class P7\_3\_transpose\_matrix {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter the number of rows of the matrix: ");  int rows = scanner.nextInt();  System.out.print("Enter the number of columns of the matrix: ");  int cols = scanner.nextInt();  int[][] matrix = new int[rows][cols];  // Input matrix elements  System.out.println("Enter the elements of the matrix:");  for (int i = 0; i < rows; i++) {  for (int j = 0; j < cols; j++) {  matrix[i][j] = scanner.nextInt();  }  }  System.out.println("Original Matrix:");  printMatrix(matrix);  // Transpose the matrix  int[][] transposeMatrix = findTranspose(matrix);  System.out.println("Transpose of the Matrix:");  printMatrix(transposeMatrix);  scanner.close();  }  // Function to find the transpose of a matrix  public static int[][] findTranspose(int[][] matrix) {  int rows = matrix.length;  int cols = matrix[0].length;  int[][] transpose = new int[cols][rows];  for (int i = 0; i < rows; i++) {  for (int j = 0; j < cols; j++) {  transpose[j][i] = matrix[i][j];  }  }  return transpose;  }  // Function to print a matrix  public static void printMatrix(int[][] matrix) {  for (int[] row : matrix) {  for (int num : row) {  System.out.print(num + "\t");  }  System.out.println();  }  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>  Enter the number of rows of the matrix: 2  Enter the number of columns of the matrix: 2  Enter the elements of the matrix:  4  5  6  7  Original Matrix:  4 5  6 7  Transpose of the Matrix:  4 6  5 7  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |

|  |  |
| --- | --- |
| **Program 1.4** | **Date: 05/01/2024** |
| **Write a Java program to find the second smallest element in an array.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.\*;  public class P7\_1\_palindrome {  public static void main(String args[]) {  Scanner s = new Scanner(System.in);  int i;  boolean flag = true;  try {  System.out.println("\nEnter a string to check for palindrome : ");  String str = s.nextLine();  str = str.toLowerCase(); // coverts entered string to lowercase  // Comparing one character at a time till middle of the string is reached  for (i = 0; i < (str.length() / 2); i++) {  if (str.charAt(i) != str.charAt(str.length() - i - 1)) {  flag = false;  break;  }  }  if (flag) {  System.out.println("\nstring " + str + " is Palindrome.");  } else {  System.out.println("\nstring " + str + " is Not Palindrome.");  }  } catch (Exception e) {  System.out.println("\nError : " + e);  }  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>    Enter the number of elements in the array: 5  Enter the elements of the array:  4  3  7  8  1  The second smallest element in the array is: 3  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |

|  |  |
| --- | --- |
| **Program 1.5** | **Date: 06/01/2024** |
| **Write a Java program to check whether a number is prime or not.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.Scanner;  public class P7\_5\_prime {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.print("Enter a number to check if it's prime: ");  int number = scanner.nextInt();  if (isPrime(number)) {  System.out.println(number + " is a prime number.");  } else {  System.out.println(number + " is not a prime number.");  }  scanner.close();  }  public static boolean isPrime(int number) {  if (number <= 1) {  return false;  }  if (number <= 3) {  return true;  }  if (number % 2 == 0 || number % 3 == 0) {  return false;  }  for (int i = 5; i \* i <= number; i += 6) {  if (number % i == 0 || number % (i + 2) == 0) {  return false;  }  }  return true;  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>  Enter a number to check if it's prime: 7  7 is a prime number.  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |

|  |  |
| --- | --- |
| **Program 1.6** | **Date: 06/01/2024** |
| **Write a java program to demonstrate Bitwise logical operators, left shift and right shift operators.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.\*;  public class P7\_6\_bitwise {  public static void main(String[] args) {  Scanner s = new Scanner(System.in);  try {  System.out.println("\nEnter the first integer : ");//Eg:12 -> Binary: 1100  int num1 = s.nextInt();  System.out.println("\nEnter the second integer : ");//Eg:7 -> Binary: 0111  int num2 = s.nextInt();  // Bitwise AND operator (&)  int resultAnd = num1 & num2; // Result: 4 (Binary: 0100)  System.out.println("Bitwise AND of " + num1 + " and " + num2 + " is: " + resultAnd);  // Bitwise OR operator (|)  int resultOr = num1 | num2; // Result: 15 (Binary: 1111)  System.out.println("Bitwise OR of " + num1 + " and " + num2 + " is: " + resultOr);  // Bitwise XOR operator (^)  int resultXor = num1 ^ num2; // Result: 11 (Binary: 1011)  System.out.println("Bitwise XOR of " + num1 + " and " + num2 + " is: " + resultXor);  // Bitwise NOT operator (~)  int resultNotNum1 = ~num1; // Result: -13 (Binary: 11111111 11111111 11111111 11110011)  System.out.println("Bitwise NOT of " + num1 + " is: " + resultNotNum1);  // Left shift operator (<<)  int resultLeftShift = num1 << 2; // Result: 48 (Binary: 110000)  System.out.println("Left shift of " + num1 + " by 2 is: " + resultLeftShift);  // Right shift operator (>>)  int resultRightShift = num2 >> 2; // Result: 1 (Binary: 0001)  System.out.println("Right shift of " + num2 + " by 2 is: " + resultRightShift);  } catch (Exception e) {  System.out.println("\nError : " + e);  }  s.close();  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>  Enter the first integer :  12  Enter the second integer :  7  Bitwise AND of 12 and 7 is: 4  Bitwise OR of 12 and 7 is: 15  Bitwise XOR of 12 and 7 is: 11  Bitwise NOT of 12 is: -13  Left shift of 12 by 2 is: 48  Right shift of 7 by 2 is: 1  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |

|  |  |
| --- | --- |
| **Program 1.7** | **Date: 08/01/2024** |
| **Write a java program to demonstrate Bitwise logical operators, left shift and right shift operators.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.Scanner;  public class P7\_7\_roots\_of\_quadratic {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.println("Enter the coefficients of the quadratic equation (ax^2 + bx + c = 0):");  System.out.print("Enter the coefficient a: ");  double a = scanner.nextDouble();  System.out.print("Enter the coefficient b: ");  double b = scanner.nextDouble();  System.out.print("Enter the coefficient c: ");  double c = scanner.nextDouble();  double discriminant = b \* b - 4 \* a \* c;  if (discriminant > 0) {  double root1 = (-b + Math.sqrt(discriminant)) / (2 \* a);  double root2 = (-b - Math.sqrt(discriminant)) / (2 \* a);  System.out.println("Roots are real and different.");  System.out.println("Root 1 = " + root1);  System.out.println("Root 2 = " + root2);  } else if (discriminant == 0) {  double root = -b / (2 \* a);  System.out.println("Roots are real and equal.");  System.out.println("Root = " + root);  } else {  double realPart = -b / (2 \* a);  double imaginaryPart = Math.sqrt(-discriminant) / (2 \* a);  System.out.println("Roots are complex and different.");  System.out.println("Root 1 = " + realPart + " + " + imaginaryPart + "i");  System.out.println("Root 2 = " + realPart + " - " + imaginaryPart + "i");  }  scanner.close();  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>  Enter the coefficients of the quadratic equation (ax^2 + bx + c = 0):  Enter the coefficient a: 1  Enter the coefficient b: -3  Enter the coefficient c: 2  Roots are real and different.  Root 1 = 2.0  Root 2 = 1.0  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |

|  |  |
| --- | --- |
| **Program 2.1** | **Date: 08/01/2024** |
| **Write a java program to demonstrate Bitwise logical operators, left shift and right shift operators.** | |

**PROGRAM :**

|  |
| --- |
| import java.util.Scanner;  public class P7\_7\_roots\_of\_quadratic {  public static void main(String[] args) {  Scanner scanner = new Scanner(System.in);  System.out.println("Enter the coefficients of the quadratic equation (ax^2 + bx + c = 0):");  System.out.print("Enter the coefficient a: ");  double a = scanner.nextDouble();  System.out.print("Enter the coefficient b: ");  double b = scanner.nextDouble();  System.out.print("Enter the coefficient c: ");  double c = scanner.nextDouble();  double discriminant = b \* b - 4 \* a \* c;  if (discriminant > 0) {  double root1 = (-b + Math.sqrt(discriminant)) / (2 \* a);  double root2 = (-b - Math.sqrt(discriminant)) / (2 \* a);  System.out.println("Roots are real and different.");  System.out.println("Root 1 = " + root1);  System.out.println("Root 2 = " + root2);  } else if (discriminant == 0) {  double root = -b / (2 \* a);  System.out.println("Roots are real and equal.");  System.out.println("Root = " + root);  } else {  double realPart = -b / (2 \* a);  double imaginaryPart = Math.sqrt(-discriminant) / (2 \* a);  System.out.println("Roots are complex and different.");  System.out.println("Root 1 = " + realPart + " + " + imaginaryPart + "i");  System.out.println("Root 2 = " + realPart + " - " + imaginaryPart + "i");  }  scanner.close();  }  } |

**OUTPUT :**

|  |
| --- |
| PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)>  Enter the coefficients of the quadratic equation (ax^2 + bx + c = 0):  Enter the coefficient a: 1  Enter the coefficient b: -3  Enter the coefficient c: 2  Roots are real and different.  Root 1 = 2.0  Root 2 = 1.0  PS E:\MCA\SEM 2\JAVA PROGRAMMING (MCA202)> |