

MUHAMMAD MUJTABA SP22-BSE-036  
PROGRAMMING FUNDAMENTAL  
SIR RIZWAN RASHID  
2<sup>ND</sup> SEMESTER  
CUI ISB  
DEC 17, 2022  
LAB ASSIGNMENT 3

## LAB 7: CODE

```
import java.util.Scanner;

public class Lab7Recursion {

    // ++++++
    // ++++++
    // ++++++
    // ++++++

    static void question9(){
        Scanner input = new Scanner(System.in);
        System.out.print("Enter a real number: ");
        int num = input.nextInt();
        System.out.print("Enter power: ");
        int power = input.nextInt();
        System.out.println("Result is " + power (num, power));
    }
    public static int power(int n, int p){
        if ( p == 0) return 1;
        else return n * power(n , p-1);
    }

    // ++++++
    // ++++++
    // ++++++
    // ++++++

    static void question10(){
        Scanner input = new Scanner(System.in);
        System.out.print("Enter non negative number to print pattern 1: ");
        int n1 = input.nextInt();
        printPattern1(n1, n1+1);
    }
}
```

```

        System.out.println();
        System.out.print("Enter non negative number to print pattern 2: ");
        int n2 = input.nextInt();
        printPattern2(n2, n2+1);
        System.out.println();
        System.out.print("Enter non negative number to print pattern 3: ");
        int n3 = input.nextInt();
        printPattern3(n3, n3+1);
        System.out.println();
        System.out.print("Enter non negative number to print pattern 4: ");
        int n4 = input.nextInt();
        printPattern4(n4);
//Q10PartA
        System.out.println();
        System.out.println("Question 10 Part A");
        System.out.println("Enter sequence of integers: ");
        int seq = input.nextInt();
        System.out.println("Reverse order is ");
        reverse(seq);
//Q10PartB
        System.out.println();
        System.out.println("Question 10 Part B");
        System.out.println("Enter integer number to find binary: ");
        int num = input.nextInt();
        System.out.println("Binary number of " + num + " is " +
binary(num));
//Q10PartC
        System.out.println();
        System.out.println("Question 10 Part C");
        int array[] = {5, 10, 9, 8, 1};
        System.out.println("Enter integer to find index in Array ");
        int x = input.nextInt();
        int low = 0;
        int binsearch = binarySearch(array, low, array.length - 1, x);
        if (binsearch == -1) System.out.println(x + " is not present");
        else System.out.println("Index of " + x + " is " + binsearch);
    }
//Pattern1
    public static void printPattern1(int n, int x){
        if(n>=1){
            printSpace1(n);
            printStar1(n, x);
            System.out.println();
            printPattern1(n-1, x);}
    }
    public static void printSpace1(int n){
        if ((n-1)>=1){System.out.print(" ");printSpace1(n-1);}
    }

```

```

public static void printStar1(int n, int x){
    if(x-n>=1){System.out.print("* ");printStar1(n+1, x);}
}

//Pattern2
public static void printPattern2(int n, int x){
    if(n>=1){
        printSpace2(n);
        printStar2(n, x);
        System.out.println();
        printPattern2(n-1, x);}
}
public static void printSpace2(int n){if (n>=1){System.out.print("
");printSpace2(n-1);}}
public static void printStar2(int n, int x){if(x-
n>=1){System.out.print("*"); printStar2(n+1, x);}}

//Pattern3
public static void printPattern3(int n, int x){
    if (n>0){printStar3(n, x);
        System.out.println();
        printPattern3(n-1, x);}
}
public static void printStar3(int n, int x){
    if (x-n>=1){System.out.print("*");printStar3(n+1, x);}
}

//Pattern 4
public static void printPattern4(int n){
    if (n>0){
        printStar4(n);
        System.out.println();
        printPattern4(n-1);}
}
public static void printStar4(int n){
    if (n>0){System.out.print("*");printStar4(n-1);}
}

//Method Part A reverse
public static void reverse(int seq){
    if (seq < 10){System.out.println(seq);return; }
    else {System.out.print(seq % 10);reverse(seq / 10);}}

//Method Part B Binary
public static int binary(int num){
    if (num == 0) return 0;
    else return (num % 2 + 10 * binary(num / 2));
}

```

```

//Method Binary Search
public static int binarySearch(int array[], int low, int leng, int x){
    if (leng >= low){
        int mid = low + (leng - 1) / 2;
        if (array[mid] == x) return mid;
        if (array[mid] > x) return binarySearch(array, low, mid - 1, x);
        //present in left side
        return binarySearch(array, mid + 1, leng, x); //present in right
side
    }
    return -1;
}
// ++++++
// ++++++
// ++++++
// ++++++

public static void main(String[] args){
    question9();
    question10();
}
}

```

## LAB 8 : CODE

```

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

public class Lab8 {

    public void rateFood(){
        int[] totalRatings = new int[10];
        Scanner s = new Scanner(System.in);
        System.out.print("Rate the quality of food in your cafe (1-10).");
        for(int n = 0; n < 10; n++){
            System.out.print("Response . " + n + ": ");
            int rating = s.nextInt();
            totalRatings[n] = rating;
            System.out.print("\n");
        }
        int summary = 0;
        for(int n = 0; n < 10; n++){ summary += totalRatings[n]; }
        summary /= 10;
        System.out.print("Average food quality: ");
        System.out.print((summary > 5) ? "tastes good -)" : "awful :-(");
    }
}

```

```

public void modify(int[] arr){
    for(int i = 0; i < arr.length(); i++) arr[i] *= 3;
}

public void reverseCopy(int[] from, int[] to){
    if(from.length() != to.length()) return ;
    for(int i = 0; i < from.length(); i++){
        to[to.length() - i] = from[i];
    }
}

public static void main(String[] args){
    rateFood();
    int[] arr = new int[10]; modify(arr);
    int[] A = {1,2,3}; int[] B = new int[3];
    reverseCopy(A, B);
    System.out.print("'" + B[0] + " " + B[1] + " " + B[2]);
}
}

```

## LAB 10: CODE

### QUESTION .1

```

class Lab10_Question1 {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);
        int[][] arr = new int[3][4];
        for (int i = 0; i < 3; i++)
            for (int j = 0; j < 4; j++){
                System.out.println("Enter element at " + i + "X" + j + "
in Matrix");
                arr[i][j] = input.nextInt();}
        System.out.println("Matrix is ");
        for(int i=0; i < arr.length; i++) {
            for(int j=0; j < arr[i].length; j++)
                System.out.print(arr[i][j] + " ");
            System.out.println(); }
        int row1 = rowPSum(arr, 0);
        int row2 = rowPSum(arr, 1);
        int row3 = rowPSum(arr, 2);
        int col1 = colPSum(arr, 0);
        int col2 = colPSum(arr, 1);
        int col3 = colPSum(arr, 2);
        int col4 = colPSum(arr, 3);
        if ((row1>row2) && (row1>row3))
            System.out.println("Sum of Prime numbers in Row 1 is Max");
    }
}

```

```

else if ((row2>row1) && (row2>row3))
    System.out.println("Sum of Prime numbers in Row 2 is Max");
else System.out.println("Sum of Prime numbers in Row 3 is Max");
if ((col1>col2) && (col1>col3) && (col1>col4))
    System.out.println("Sum of Prime numbers in Column 1 is Max");
else if ((col2>col1) && (col2>col3) && (col2>col4))
    System.out.println("Sum of Prime numbers in Column 2 is Max");
else if ((col3>col1) && (col3>col2) && (col3>col4))
    System.out.println("Sum of Prime numbers in Column 3 is Max");
else System.out.println("Sum of Prime numbers in Column 4 is
Max");
}
}
public static int rowPSum(int arr[][], int row){
    int sum = 0;
    for (int i = 0; i < 4; i++) {
        for(int j = 2 ; j <= arr[row][i]/2; j++){
            if (arr[row][i] % j == 0) break;
            else sum = sum + arr[row][i]; }
    }
    return sum;
}
public static int colPSum(int arr[][], int col){
    int sum = 0;
    for (int i = 0; i < 3; i++) {
        for(int j = 2; j <= arr[i][col]/2; j++){
            if(arr[i][col] % j == 0) break;
            else sum = sum + arr[i][col]; }
    }
    return sum;
}
}
}

```

## QUESTION.2

```

import java.util.*;
public class Lab10_Question2{
    public static void main(String[]args){
        Scanner input=new Scanner(System.in);
        int[][] array= new int [3] [4];
        int max=0;int row=0;int col=0;int max2=0;
        for (int rows=0; rows<array.length;rows++){
            int count=0;
            System.out.println("Enter elements of row"+(rows+1));
            for (int cols=0; cols< array[0].length; cols++){
                array [rows][cols]=input.nextInt();
                if (array[rows][cols]%2==0) count++;
                if(count>max){max=count; row=rows+1;}
            }
        }
    }
}

```

```

    }
}
for (int cols=0; cols<array[0].length;cols++){
    int count2=0;
    for (int rows=0; rows< array.length; rows++){
        if (array[rows][cols]%2==0) count2++;
        if(count2>max2){max2=count2;col=cols+1;}}
    }
print2D(array);
if (max>max2)System.out.println("Row having max prime no is: " +
row);
    else System.out.println("col having max prime no is: " + col);
}
public static void print2D(int mat[][]){
    for (int[] row : mat)
        System.out.println(Arrays.toString(row));
}
}

```

### QUESTION .3

```

import java.util.*;

public class Lab10_Question3 {

    public static void main(String args[]) {
        int row1, col1, row2, col2;
        Scanner s = new Scanner(System.in);
        // Input dimensions of First Matrix: A
        System.out.print("Enter number of rows in first matrix: ");
        row1 = s.nextInt();
        System.out.print("Enter number of columns in first matrix: ");
        col1 = s.nextInt();
        // Input dimensions of second matrix: B
        System.out.print("Enter number of rows in second matrix: ");
        row2 = s.nextInt();
        System.out.print("Enter number of columns in second matrix: ");
        col2 = s.nextInt();
        // Requirement check for matrix multiplication
        if (col1 != row2) {
            System.out.println("Matrix multiplication is not possible");
            return;
        }
        int a[][] = new int[row1][col1];
        int b[][] = new int[row2][col2];
        int c[][] = new int[row1][col2];
    }
}

```

```

// Input the values of matrices
System.out.println("\nEnter values for matrix A : ");
for (int i = 0; i < row1; i++) {
    for (int j = 0; j < col1; j++) a[i][j] = s.nextInt();
}
System.out.println("\nEnter values for matrix B : ");
for (int i = 0; i < row2; i++) {
    for (int j = 0; j < col2; j++) b[i][j] = s.nextInt();
}
// Perform matrix multiplication
// Using for loop
System.out.println("\nMatrix multiplication is : ");
for (int i = 0; i < row1; i++) {
    for (int j = 0; j < col2; j++) {
        // Initialize the element C(i,j) with zero
        c[i][j] = 0;
        // Dot product calculation
        for (int k = 0; k < col1; k++) {
            c[i][j] += a[i][k] * b[k][j];
        }
        System.out.print(c[i][j] + " ");
    }
    System.out.println();
}
}
}
}

```