JAVA PROGRAMMING. DIT 409.

ASSIGNMENT ON METHODS.

QUESTIONS.

1. Create a java project, name it methods_in_java, in the project create a package named java_methods and in the package, create a class and named methods.

b. in the classmethods, write a method that asks user to enter three numbers, using if statement determine the largest number, the smallest number and display the results in the following format.
The smallest number: ?
The largest number number: ?
? is your largest and ? smallest number.

```
package java_methods;

import java.util.Scanner;

public class methods {
    public static void main(String[] args) {
        findLargestAndSmallest();
    }

    public static void findLargestAndSmallest() {
        Scanner input = new Scanner(System.in);

        System.out.println("Please enter three numbers:");
        int num1 = input.nextInt();
        int num2 = input.nextInt();
        int num3 = input.nextInt();

        int smallest = num1;
        if (num2 < smallest) {
            smallest = num2;
        }
        if (num3 < smallest) {
            smallest = num3;
        }

        int largest = num1;
        if (num2 > largest) {
            largest = num2;
        }
        if (num3 > largest) {
            largest = num3;
        }

        System.out.println("The smallest number: " + smallest);
        System.out.println("The largest number: " + largest);
        System.out.println(smallest + " is your smallest number and " + largest + " is your largest number.");

        input.close();
    }
}
```

2. Create a java project, package and class. In the class, write a method that asks a lecturer to enter marks for three units, java programming, networking and maths. The method should compute the average marks for three units and output the data in the following format.

```
marks for java programming is: ?
marks for networking is: ?
marks for maths is: ?
the average is: ?
```

```
import java.util.Scanner;
```

```
class MarksCalculator {
    public static void main(String[] args) {
        calculateAverageMarks();
    }

    public static void calculateAverageMarks() {
        Scanner input = new Scanner(System.in);

        System.out.println("Please enter the marks for Java Programming:");
        int javaMarks = input.nextInt();

        System.out.println("Please enter the marks for Networking:");
        int networkingMarks = input.nextInt();

        System.out.println("Please enter the marks for Maths:");
        int mathsMarks = input.nextInt();

        double average = (javaMarks + networkingMarks + mathsMarks) / 3.0;

        System.out.println("Marks for Java Programming is: " + javaMarks);
        System.out.println("Marks for Networking is: " + networkingMarks);
        System.out.println("Marks for Maths is: " + mathsMarks);
        System.out.println("The average is: " + average);

        input.close();
    }
}
```

3. ***QUESTION 3:***

Write a method that asks user to enter the year, the program should check if the year is a leap year, and output the text the year you entered is a leap year.

Create a java project, a package and a class, in the class, write a program to calculate the area of a triangle. The program should have thee non-static methods:

One of the methods should ask the user to enter the base and the height of a triangle.

The other method should compute the area of the rectangle.

The other method should output the calculated area of the triangle an display it to the user.

```
import java.util.Scanner;


public class LeapYearChecker {
    public static void main(String[] args) {
        checkLeapYear();
    }


    public static void checkLeapYear() {
        Scanner input = new Scanner(System.in);


        System.out.println("Please enter the year:");
        int year = input.nextInt();


        if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
            System.out.println("The year you entered, " + year + ", is a leap year.");
        } else {
            System.out.println("The year you entered, " + year + ", is not a leap year.");
        }


        input.close();
    }
}
```

4. Create a java program that has one non-static method, two static methods and a constructor.

```
public class ProgramWithMethods {
    // Constructor
    public ProgramWithMethods() {
        System.out.println("Constructor called. Object created.");
    }

    // Non-static method
    public void nonStaticMethod() {
        System.out.println("This is a non-static method.");
    }

    // Static method 1
    public static void staticMethod1() {
        System.out.println("This is static method 1.");
    }

    // Static method 2
    public static void staticMethod2() {
        System.out.println("This is static method 2.");
    }

    public static void main(String[] args) {
        // Creating an object of the class to access non-static method and constructor
        ProgramWithMethods program = new ProgramWithMethods();

        // Calling the non-static method using the object
        program.nonStaticMethod();

        // Calling the static methods directly using class name
        ProgramWithMethods.staticMethod1();
        ProgramWithMethods.staticMethod2();
    }
}
```

Question one: [15 marks]

a. A prime number is a number that is evenly divisible only by itself and 1. For example, the number 5 is prime because it can be evenly divided only by 1 and 5. The number 6, however, is not prime because it can be divided evenly by 1, 2, 4, and 6.
   Write a method named isPrime, which takes an integer as an argument and returns true if the argument is a prime number, or false otherwise. Also write main method that displays prime numbers between 1 to 500.

```
public class PrimeNumbers {

    // Method to check if a number is prime

    public static boolean isPrime(int number) {

        if (number <= 1) {

            return false;

        }

        for (int i = 2; i <= Math.sqrt(number); i++) {

            if (number % i == 0) {

                return false;

            }

        }

        return true;

    }


    public static void main(String[] args) {

        System.out.println("Prime numbers between 1 to 500:");

        for (int i = 2; i <= 500; i++) {

            if (isPrime(i)) {

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

            }

        }

    }

}
```

b. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be: 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

```
public class FibonacciSequence {
    public static void main(String[] args) {
        int n = 10; // Number of terms to display
        int firstTerm = 1;
        int secondTerm = 2;

        System.out.println("The first " + n + " terms of the Fibonacci sequence:");

        System.out.print(firstTerm + " "); // Display the first term
        System.out.print(secondTerm + " "); // Display the second term

        for (int i = 3; i <= n; i++) {
            int nextTerm = firstTerm + secondTerm;
            System.out.print(nextTerm + " "); // Display the next term
            firstTerm = secondTerm;
            secondTerm = nextTerm;
        }
    }
}
```

c. By considering the terms in the Fibonacci sequence whose values do not exceed four million, write a Java method to find the sum of all the even- valued terms.

```
public class EvenValuedFibonacciSum {

    public static void main(String[] args) {

        int limit = 4000000; // Limit for the Fibonacci sequence

        int sum = 0;

        int m = 1;

        int n = 2;


        while (n <= limit) {

            if (n % 2 == 0) {

                sum += n; // Add the even-valued term to the sum

            }

            int nextTerm = m + n;

            m = n;

            n = nextTerm;

        }


        System.out.println("The sum of even-valued terms in the Fibonacci sequence not exceeding 4,000,000 is: " + sum);

    }

}
```

Question two: [15 marks]

A palindrome number is a number that remain the same when read from behind or front (a number that is equal to reverse of number) for example, 353 is palindrome because reverse of 353 is 353 (you see the number remains the same). But a number like 591 is not palindrome because reverse of 591 is 195 which is not equal to 591. Write Java program to check if a number entered by the user is palindrome or not. You should provide the user with a GUI interface to enter the number and display the results on the same interface.

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;


public class PalindromeChecker extends JFrame {

    private JTextField inputField;

    private JLabel resultLabel;


    public PalindromeChecker() {

        setTitle("Palindrome Checker");

        setSize(300, 150);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


        JPanel panel = new JPanel();

        panel.setLayout(new GridLayout(3, 1));


        JLabel inputLabel = new JLabel("Enter a number:");

        inputField = new JTextField();

        JButton checkButton = new JButton("Check Palindrome");

        resultLabel = new JLabel();


        checkButton.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                String input = inputField.getText();

                if (isPalindrome(input)) {
```

```java
                    resultLabel.setText(input + " is a palindrome!");
                } else {
                    resultLabel.setText(input + " is not a palindrome.");
                }
            }
        });

        panel.add(inputLabel);
        panel.add(inputField);
        panel.add(checkButton);
        panel.add(resultLabel);

        add(panel);
        setVisible(true);
    }

    private boolean isPalindrome(String input) {
        String reversed = new StringBuilder(input).reverse().toString();
        return input.equals(reversed);
    }

    public static void main(String[] args) {
        new PalindromeChecker();
    }
}
```

Write a Java program that takes 15 values of type integer as inputs from user, store the values in an array.

a) Print the values stored in the array on screen

```
import java.util.Scanner;


public class ArrayPrinter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] numbers = new int[15];


        System.out.println("Enter 15 integer values:");


        for (int i = 0; i < 15; i++) {
            numbers[i] = scanner.nextInt();
        }


        System.out.println("Values stored in the array:");
        for (int number : numbers) {
            System.out.print(number + " ");
        }
    }
}
```

b) Ask user to enter a number, check if that number (entered by user) is present in array

or not. If it is present print, “the number found at index (index of the number) ” and the text “number not found in this array”

```
import java.util.Scanner;

public class ArraySearchWithIndex {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int[] numbers = new int[15];

        System.out.println("Enter 15 integer values:");

        for (int i = 0; i < 15; i++) {
            numbers[i] = scanner.nextInt();
        }

        System.out.println("Enter a number to search in the array:");
        int searchNumber = scanner.nextInt();

        int index = -1;
        for (int i = 0; i < 15; i++) {
            if (numbers[i] == searchNumber) {
                index = i;
                break;
            }
        }

        if (index != -1) {
            System.out.println("The number found at index " + index);
        } else {
            System.out.println("Number not found in this array");
        }
    }
}
```

c) Create another array, copy all the elements from the existing array to the new array but in reverse order. Now print the elements of the new array on the screen

```
import java.util.Arrays;

public class ArrayReverseCopy {
    public static void main(String[] args) {
        int[] originalArray = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15};
        int[] reverseArray = new int[15];

        for (int i = 0; i < originalArray.length; i++) {
            reverseArray[i] = originalArray[originalArray.length - i - 1];
        }

        System.out.println("Elements of the new array in reverse order:");
        for (int number : reverseArray) {
            System.out.print(number + " ");
        }
    }
}
```

d) Get the sum and product of all elements of your array. Print product and the sum each on its own line.

```
public class ArraySumProduct {
    public static void main(String[] args) {
        int[] numbers = {3, 7, 9, 2, 8, 6, 5, 1, 4, 10, 15, 12, 11, 13, 14};

        int sum = 0;
        int product = 1;

        for (int number : numbers) {
            sum += number;
            product *= number;
        }

        System.out.println("Sum of all elements in the array: " + sum);
        System.out.println("Product of all elements in the array: " + product);
    }
}
```