Rajalakshmi Engineering College

Name: Himesh Niranjhan A
Email: 240701192@rajalakshmi.edu.in
Roll no: 240701192
Phone: 9444103224
Branch: REC
Department: CSE - Section 10
Batch: 2028
Degree: B.E - CSE

Scan to verify results

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 1_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Mandy is working on a cybersecurity project that involves basic encryption techniques. She wants to write a program that takes an integer number and performs a bitwise XOR operation to flip all the bits.

Help Mandy in this encryption using bitwise operations.

Input Format

The input consists of an integer N, representing the number to be flipped.

Output Format

The output displays "Result: " followed by an integer representing the result of the bitwise XOR operation to flip all the bits.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 0
Output: Result: 255

Answer

```
import java.util.Scanner;

class BitFlipper {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int N = sc.nextInt();

        int result = N ^ 255;

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

Status : Correct ***Marks : 10/10***

2. Problem Statement:

Tom is tasked with writing a program that determines whether a given integer is the square of another integer. A perfect square is a number that can be expressed as the square of an integer. The program should take an integer as input and determine if it is a perfect square or not.

The task is to implement the logic to check if the provided integer is the square of an integer and return the result.

Input Format

The first line of the input contains an integer, "input", where |input| represents the absolute value of the integer.

Output Format

The output should display a boolean value, "result," which should be set to true if the input is a perfect square (the square of an integer), and false if it is not.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 16

Output: Is the integer a perfect square? true

Answer

```
import java.util.Scanner;

class PerfectSquareChecker {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int input = sc.nextInt();

        boolean result = isPerfectSquare(input);
        System.out.println("Is the integer a perfect square? " + result);
    }

    public static boolean isPerfectSquare(int n) {
        if (n < 0) return false;
        int sqrt = (int)Math.sqrt(n);
        return sqrt * sqrt == n;
    }
}
```

Status : Correct ***Marks : 10/10***

3. Problem Statement:

"Write a program that helps identify the type of a triangle based on the lengths of its three sides. The program prompts the user to input the lengths of sides 'a,' 'b,' and 'c,' and then it classifies the triangle as 'Equilateral' if all sides are equal, 'Isosceles' if two sides are equal, or 'Scalene' if all sides are different. Can you provide the Java code for this

task?"

Input Format

The first line of the input is an integer 'a' representing the length of side 'a.'

The second line of the input is an integer 'b' representing the length of side 'b.'

The third line of the input is an integer 'c' representing the length of side 'c.'

Output Format

The program outputs a single line that specifies the type of the triangle: "Equilateral," "Isosceles," or "Scalene."

Sample Test Case

Input: 3
4
5
Output: The triangle is Scalene

Answer

```
// You are using Java
import java.util.Scanner;

class TriangleTypeIdentifier {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);


        int a = sc.nextInt();
        int b = sc.nextInt();
        int c = sc.nextInt();

        String type;

        if (a == b && b == c) {
            type = "Equilateral";
        } else if (a == b || b == c || a == c) {
            type = "Isosceles";
        } else {
            type = "Scalene";
        }
```

```
        System.out.println("The triangle is " + type);
    }
}
```

Status : Correct **Marks : 10/10**

4. Problem Statement

In a logistics company, each delivery pack contains a specific number of items, and the priority customer receives double the amount. Write a program to determine the total number of delivery packs required for the operation, considering the number of items per pack and the number of customers given as input by the user.

Example

Input:

Number of items per pack = 96

Number of customers = 8

Output:

10

Explanation:

Given the number of items per pack = 96 and the number of customers = 8, the calculations are as follows:

Total number of items needed = number of items per pack * number of customers = 96 * 8 = 768.Priority customer's share = double the amount of items per pack = 2 * 96 = 192.Total items with the priority customer = total items needed + priority share = 768 + 192 = 960.Number of packs needed = (960 + 96 − 1)/ 96 = 10.98Since we cannot have a fraction of a pack, the output is 10.

Input Format

The input consists of two space-separated integers N and C, representing the number of items per pack and the number of customers.

Output Format

The output displays an integer, representing the total number of delivery packs required for the operation.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1 1

Output: 3

Answer

```
import java.util.Scanner;

class DeliveryPackCalculator {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int N = sc.nextInt();
        int C = sc.nextInt();

        int totalItems = N * C;

        int priorityShare = 2 * N;

        int allItems = totalItems + priorityShare;

        int packs = (allItems + N - 1) / N;

        System.out.println(packs);
    }
}
```

Status : Correct ***Marks : 10/10***