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_PAH

Attempt : 1
Total Mark : 40
Marks Obtained : 30

Section 1 : Coding

1. PROBLEM STATEMENT:

Maria, a software developer, is working on a program to determine if two given integers which can be either positive or negative integers have the same parity (both even or both odd). She needs your help in writing this program.

Write a program that takes two integers as input and checks if both integers are either even or odd.

Input Format

The input consists of two lines:

The first line consists of an integer (input1) which can be either positive or negative.

The second line consists of an integer (input2) which can be either positive or negative.

Output Format

The output is dispalyed in the following format:

If both integers have the same parity (i.e., both even or both odd), print:

"Both integers are either even or odd"

Otherwise, print:

"The integers have different parities"

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 2
-4
Output: Both integers are either even or odd

Answer

```
import java.util.Scanner;

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

        int input1 = scanner.nextInt();
        int input2 = scanner.nextInt();

        if (input1 >= -1000 && input1 <= 1000 && input2 >= -1000 && input2 <= 1000)
{
```

```
            if (input1 % 2 == input2 % 2) {
                System.out.println("Both integers are either even or odd");
            } else {
                System.out.println("The integers have different parities");
            }
        } else {
            System.out.println("Inputs must be in the range -1000 to 1000");
        }

        scanner.close();
    }
}
```

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

2. PROBLEM STATEMENT:

Maria, a software developer, is working on a project to create a simple program to determine which of two integers is closest to zero. The integers can be either positive or negative. The program needs to take two integer inputs and calculate which one is closer to zero. If both integers are equidistant from zero, the program should return 0.

Input Format

The input contains two lines:

The first line of the input contains an integer, which can be either a positive or a negative integer.

The second line of the input contains an integer, which can be either a positive or a negative integer.

Output Format

The output displays the integer that is closest to zero in the following format:

"The integer closest to zero is: [closest_integer]"

Here, [closest_integer] should be replaced with the integer that is closer to zero based on its absolute value.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 5
8
Output: The integer closest to zero is: 5

Answer

```
import java.util.Scanner;

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

      int num1 = sc.nextInt();
      int num2 = sc.nextInt();

      int absNum1 = Math.abs(num1);
      int absNum2 = Math.abs(num2);

      int closest;

      if (absNum1 < absNum2) {
         closest = num1;
      } else if (absNum2 < absNum1) {
         closest = num2;
      } else {
         closest = 0;
      }

      System.out.println("The integer closest to zero is: " + closest);
   }
}
```

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

3. Problem Statement

Mickey and Miney are walking through a magical forest. The forest is full of enchanted stones, each with a unique number. There is a legend that says the magic power of the stones can be revealed by using a special operation. To determine the magic power of a given stone, you need to perform a bitwise AND operation with the number 15.

Each stone's number is represented by an integer, and Mickey needs to find the magic power of each stone by applying this operation.

Your task is to help Mickey compute the result of the bitwise AND operation of the given stone number with 15, and print the result.

Input Format

The input consists of a single integer.

Output Format

The output should display a single integer, which is the result of the bitwise AND operation between input and 15.

Refer to the sample output for format specifications.

Sample Test Case

Input: 25
Output: 9

Answer

```
import java.util.Scanner;

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

```
        int magicPower = stoneNumber & 15;

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

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

4. Problem Statement

In the Kingdom of Delivery Logistics, there is a giant truck used for transporting packages across the kingdom. The truck has a maximum capacity represented by an integer, and each package also has a specific weight. The truck’s efficiency and safety depend on whether the weight of the package is below a certain threshold.

The kingdom's delivery service has a rule: if the weight of a package is less than one-third of the truck’s total capacity, the package is eligible for quick processing and dispatch. However, if the weight is too heavy, the package will require special handling.

As a logistics manager, you need to check whether the weight of the package is less than one-third of the truck's total capacity.

Write a program using a ternary operator that helps determine whether the package weight meets the requirement for quick processing or if it needs special handling.

Input Format

The first line of input consists of an integer p, representing the weight of the package.

The second line consists of an integer w, representing the total weight capacity of the truck.

Output Format

The first line of output prints "One-third of Truck: X," where X is one-third of the truck's total weight capacity as a double value with two decimal places.

The second line of output displays one of the following:

1. If p is less than one-third of the truck's total weight capacity, print "Package weight is less than one-third of the truck's capacity".
2. Otherwise, print "Package weight is not less than one-third of the truck's capacity".

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 13
60

Output: One-third of Truck: 20.00
Package weight is less than one-third of truck's capacity

Answer

```
import java.util.Scanner;

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

        int p = sc.nextInt();
        int w = sc.nextInt();

        double oneThird = w / 3.0;

        System.out.printf("One-third of Truck: %.2f%n", oneThird);

        String result = (p < oneThird)
            ? "Package weight is less than one-third of the truck's capacity"
            : "Package weight is not less than one-third of the truck's capacity";

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

Status : Wrong ***Marks : 0/10***