Rajalakshmi Engineering College

Name: Priyan S
Email: 240701402@rajalakshmi.edu.in
Roll no: 240701402
Phone: 9150170939
Branch: REC
Department: CSE - Section 6
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:

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

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

class main{
   public static void main(String[]args){
      Scanner ip=new Scanner(System.in);
      int n=ip.nextInt();
      double m=Math.sqrt(n);

      int num=(int)m;
      boolean result=((num*num)==n);

      System.out.println("Is the integer a perfect square? "+result);


   }

}
```

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

2. 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

```
// You are using Java

import java.util.*;

class main{
   public static void main(String[]args){
      Scanner ip=new Scanner(System.in);
      int n=ip.nextInt();
      int flipped=n^255;
      System.out.println("Result:"+flipped);

   }
}
```

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

3. Problem Statement:

Gilbert is tasked with writing a program that checks whether a given integer is an odd number. An odd number is one that cannot be exactly divided by 2. The program should take an integer as input and determine if it is an odd number or not. The task is to implement the logic to check if the provided integer is odd and return the result.

Input Format

The first line of the input contains an integer, "input".

Output Format

The output should display a boolean value, "result," which should be set to true if the input integer is an odd number and false if it is even.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 0
Output: Is the integer odd? false

Answer

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

class check{
   public static void main(String[]args){
      Scanner sc=new Scanner(System.in);
      int n=sc.nextInt();
      boolean result=(n%2!=0);
      System.out.println("Is the integer odd?"+result);
   }
}
```

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

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

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

      int tot_item=n*m;
      int priority=n*2;
      int tot_customer=tot_item+priority;
      double number_pack=(tot_customer+n-1)/n;

      System.out.println((int)number_pack);
   }
}
```

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