Rajalakshmi Engineering College

Name: SUBASH R
Email: 240701538@rajalakshmi.edu.in
Roll no: 240701538
Phone: 9150202710
Branch: REC
Department: CSE - Section 7
Batch: 2028
Degree: B.E - CSE

2024_28_III_OOPS Using Java Lab

2028_REC_OOPS using Java_Week 2_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

Samantha is a diligent math student who is exploring the world of programming. She is learning Java and has recently studied conditional statements. One day, her teacher gives her an interesting problem to solve, which takes a number as input and checks whether it is a multiple of 5 or 7.

Help her complete the task.

Input Format

The input consists of a single integer N, representing the number to be checked.

Output Format

If the number is a multiple of 5 but not 7, the output prints "N is a multiple of 5".

If the number is a multiple of 7, the output prints "N is a multiple of 7".

Otherwise the output prints "N is neither multiple of 5 nor 7" where N is an entered integer.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10
Output: 10 is a multiple of 5

Answer

```java
import java.util.*;

public class Main{
    public static void main(String[]args){
        Scanner in=new Scanner(System.in);
        int n=in.nextInt();
        if(n%7==0) System.out.println(n+" is a multiple of 7");
        else if(n%5==0) System.out.println(n+" is a multiple of 5");
        else System.out.println(n+" is neither multiple of 5 nor 7");
    }
}
```

Status : Correct   *Marks : 10/10*

2. Problem Statement

Joe has a favourite number, let's call it X. He wants to check if X is divisible by the sum of its digits. If it is, he considers it a lucky number. If not, he wants to find the closest smaller number, that is divisible by the sum of digits of X. Joe has challenged his friends to solve this puzzle at his birthday party.

Example

Input:

157

Output:

157 is not divisible by the sum of its digits.

The closest smaller number that is divisible: 156

Explanation:

The sum of the digits of X is 1+5+7=13. Since 157 is not divisible by 13, we need to find the closest smaller number that is divisible by 13. 156 is divisible by 13, it is the closest smaller number that meets the requirement.

Input Format

The input consists of an integer X, representing Joe's favourite number.

Output Format

If X is a lucky number, then the output must be in the format: "X is divisible by the sum of its digits."

If not, then the output must be in the format:

"X is not divisible by the sum of its digits.

The closest smaller number that is divisible: Y",

where X is the entered number and Y is the closest number.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 120
Output: 120 is divisible by the sum of its digits.

Answer

import java.util.*;

```java
public class Main{
    public static void main(String[]args){
        Scanner in =new Scanner(System.in);
        int n=in.nextInt();
        int n1=n,sum=0;
        while(n1!=0){
            sum+=n1%10;
            n1/=10;
        }
        if(n%sum==0)
            System.out.println(n+" is divisible by the sum of its digits.");
        else{
            System.out.println(n+" is not divisible by the sum of its digits.");
            n1=n;
            while(n1%sum!=0)
                n1--;
            System.out.println("The closest smaller number that is divisible: "+n1);
        }
    }
}
```

Status : Correct   *Marks : 10/10*

3. Problem Statement

Raj is solving a physics problem involving projectile motion, where he
needs to calculate the time a ball hits the ground using a quadratic
equation of the form $ax^2 + bx + c = 0$. Depending on the coefficients, the
ball may hit the ground once, twice, or not at all in real time.

Help Raj find all real roots of the equation, if any.

Note: discriminant = $b2 - 4ac$

Input Format

The input consists of three space-separated doubles a, b, and c, representing the
coefficients of the quadratic equation.

Output Format

If there are two real roots, print:

- "Two real solutions:"
- "Root1 = <value>"
- "Root2 = <value>"

If there is one real root, print:

- "One real solution:"
- "Root = <value>"

If there are no real roots, print:

- "There are no real solutions."

Note: values are rounded to two decimal places.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1 6 9

Output: One real solution:

Root = -3.00

Answer

```java
import java.util.*;

public class Main{
    public static void main(String []args){
        Scanner in=new Scanner(System.in);
        double a=in.nextInt(),b=in.nextInt(),c=in.nextInt();
        double root1=(-b+Math.sqrt((b*b)-(4*a*c)))/(2*a);
        double root2=(-b-Math.sqrt((b*b)-(4*a*c)))/(2*a);
        if(Double.isNaN(root1) || Double.isNaN(root2))
            System.out.printf("There are no real solutions.");
        else if(root1!=root2)
            System.out.printf("Two real solutions:\nRoot1 = %.2f\nRoot2 = %.2f",root1,root2);
        else if(root1==root2)
            System.out.printf("One real solution:\nRoot = %.2f",root1);
```

```
        }
    }
```

Status : Correct   *Marks : 10/10*

4. Problem Statement

Ram wants to evaluate the time required to break even on an investment based on initial costs, monthly profits, and monthly expenses. Write a program to calculate the break-even point in months and categorize the return on investment.

Compute the break-even point by using the formula: initial cost / (monthly profit - monthly expenses)Based on the break-even point, classify the return on investment into one of the following categories:Quick Return: If the break-even point is 3 months or fewer.Average Return: If the break-even point is between 4 and 12 months, inclusive.Long-term Return: If the break-even point exceeds 12 months.

Ram is new to programming, so he seeks your assistance in creating the program.

Note: monthly profit is always greater than monthly expenses.

Input Format

The first line of input consists of a double value representing the initial cost.

The second line consists of a double value representing the monthly profit.

The third line consists of a double value representing the monthly expenses.

Output Format

The first line prints "Break-even Point:", followed by the break-even point as a decimal number (of double datatype), formatted to two decimal places.

The second line prints "Category: ",followed by the investment return as a String, which can be one of:

- "Quick Return" if break-even point ≤ 3

- "Average Return" if break-even point ≤ 12
- "Long-term Return" if break-even point > 12

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 10000.50
5000.75
1000.10

Output: Break-even Point: 2.50
Category: Quick Return

Answer

```java
import java.util.*;

public class Main{
    public static void main(String[]args){
        Scanner in=new Scanner(System.in);
        double a=in.nextDouble(),b=in.nextDouble(),c=in.nextDouble();
        double bep=a/(b-c);
        System.out.printf("Break-even Point: %.2f\n",bep);
        if(bep<=3)
            System.out.println("Category: Quick Return");
        else if(bep<=12)
            System.out.println("Category: Average Return");
        else
            System.out.println("Category: Long-term Return");
    }
}
```

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