Rajalakshmi Engineering College

Name: SONASREE RP
Email: 240701521@rajalakshmi.edu.in
Roll no: 240701521
Phone: 7305340666
Branch: REC
Department: CSE - Section 10
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

Noah is analyzing numbers within a given range [A, B] and wants to calculate a special sum. For each number in the range, he calculates the product of its odd digits (ignoring even digits). If the number contains no odd digits, it is skipped. The sum of these products for all numbers in the range is the result.

Write a program to compute this sum.

Example

Input:

10 12

Output:

3

Explanation:

For 10, odd digits = 1, product = 1.

For 11, odd digits = 1, 1, product = 1 * 1 = 1.

For 12, odd digits = 1, product = 1.

Total sum = 1 + 1 + 1 = 3

Input Format

The input consists of two space-separated integers A and B, representing the inclusive range boundaries.

Output Format

The output prints a single integer representing the sum of the products of odd digits for all numbers in the range.

Refer to the sample output for the formatting specifications.

Sample Test Case

Input: 10 12
Output: 3

Answer

```java
// You are using Java
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        int A = sc.nextInt();
        int B = sc.nextInt();
        int totalSum = 0;

        for(int i = A; i <= B; i++){
            int num = i;
            int product = 1;
            boolean hasOdd = false;
```

```
    while(num > 0){
        int digit = num % 10;
        if(digit % 2 == 1){
            hasOdd = true;
            product *= digit;
        }
        num /= 10;
    }

    if(hasOdd) totalSum += product;
    }

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

Status : Correct   *Marks : 10/10*

2. Problem Statement

Maya, a student in an arts and crafts class, wants to create a pattern using stars (*) in a specific format. She plans to use a program to help her construct the pattern.

Write a program that takes an integer as input and constructs the following pattern using nested for loops.

Input: 5

Output:

*

* *

* * *

* * * *

* * * * *

```
* * * *

* * *

* *

*
```

Input Format

The input consists of a number (integer) representing the number of rows.

Output Format

The output displays the required pattern.

Refer to the sample output for the formatting specifications.

Sample Test Case

```
Input: 5
Output: *
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

Answer

```java
// You are using Java
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        int rows = sc.nextInt();

        for(int i=1; i<=rows; i++){
            for(int j=1; j<=i; j++){
                System.out.print("* ");
```

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

    for(int i=rows-1; i>=1; i--){
        for(int j=1; j<=i; j++){
            System.out.print("* ");
        }
        System.out.println();
    }
  }
}
```

Status : Correct   *Marks : 10/10*

3. 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
// You are using Java
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        int N = sc.nextInt();

        if(N % 5 == 0 && N % 7 != 0){
            System.out.println(N + " is a multiple of 5");
        } else if(N % 7 == 0){
            System.out.println(N + " is a multiple of 7");
        } else {
            System.out.println(N + " is neither multiple of 5 nor 7");
        }
    }
}
```

Status : Correct   *Marks : 10/10*

4. 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

```java
// You are using Java
import java.util.*;
public class Main{
    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        int X = sc.nextInt();

        int sum = 0, temp = X;
        while(temp > 0){
```

```
        sum += temp % 10;
        temp /= 10;
    }

    if(X % sum == 0){
        System.out.println(X + " is divisible by the sum of its digits.");
    } else {
        System.out.println(X + " is not divisible by the sum of its digits.");
        int Y = X - 1;
        while(Y > 0 && Y % sum != 0){
            Y--;
        }
        System.out.println("The closest smaller number that is divisible: " + Y);
    }
  }
}
```

Status : Correct   *Marks : 10/10*