

Rajalakshmi Engineering College

Name: Shreenidhi T
Email: 240701503@rajalakshmi.edu.in
Roll no: 240701503
Phone: 9150942326
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 2_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. 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

```
// You are using Java
import java.util.*;
class maya{
    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.println("*");
    }
}

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

```

Status : Correct

Marks : 10/10

2. Problem Statement

Ted, the computer science enthusiast, has accepted the challenge of writing a program that checks if the number of digits in an integer matches the sum of its digits.

Guide Ted in designing and writing the code to solve this problem using a 'do-while' loop.

Input Format

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

Output Format

If the sum is equal to the number of digits, print "The number of digits in N matches the sum of its digits."

Else, print "The number of digits in N does not match the sum of its digits."

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 20

Output: The number of digits in 20 matches the sum of its digits.

Answer

// You are using Java

```
import java.util.*;
```

```
class ted{
```

```
    public static void main(String args[]){
```

```
        Scanner sc=new Scanner(System.in);
```

```
        int N=sc.nextInt();
```

```
        int sum=0;
```

```
        int digicount=0;
```

```
        int num=N;
```

```
        do{
```

```
            int digit = num%10;
```

```
            sum+=digit;
```

```
            digicount++;
```

```
            num/=10;
```

```
        }
```

```
        while(num>0);
```

```
        if(digicount==sum){
```

```
            System.out.println("The number of digits in "+N+ "matches the sum of its  
digits.");
```

```
        }
```

```
        else{
```

```
            System.out.println("The number of digits in "+N+ "does not match the sum  
of its digits.");
```

```
        }
```

```
    }  
}
```

Status : Correct

Marks : 10/10

3. 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

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

```

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

    int a = sc.nextInt();
    int b=sc.nextInt();
    int totalsum=0;
    for(int num=a;num<=b;num++){
        int temp=num;
        int product=1;
        boolean hasODD=false;

        while ( temp>0){
            int digit = temp%10;
            if(digit%2!=0){
                product*=digit;
                hasODD=true;
            }
            temp/=10;
        }
        if(hasODD){
            totalsum+=product;
        }
    }
    System.out.println(totalsum);
}
}

```

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

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

```
class joe{
    public static void main(String arg[]){
        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 cl=x-1;
            while(cl>0&&cl%sum!=0){
                cl--;
            }
            System.out.println("The closest smaller number that is divisible: "+cl);
        }
    }
}
```

Status : Correct

Marks : 10/10