Title of the Article

Disarium number can be defined as a number when the sum of its digits raised to the power of their respective positions in the number is equal to the original number.

The positions are counted from left to right of the number and the position starts from 1.

To show you some instances:

Instance-1

Input number is 518

Let’s check it by using the logic of disarium number-

51 + 12 + 83 = 5 + 1 + 512 = 518which is equal to the original number.

Hence, 518 is a disarium number.

Instance-2

Input number is 175

Let’s check it by using the logic of disarium number-

11 + 72 + 53 = 1 + 49 + 125 = 175which is equal to the original number.

Hence, 175 is a disarium number.

**Some of other examples of disarium numbers include 89, 135, 518 etc.**

Syntax:

To get the length of the Integer, we will use the Java String class inbuilt **length()** method which returns the length of the String object.

Means first we will convert integer value to String value by using inbuilt  **toString()** method.

Following is the Syntax to get total number of digits in a number by converting the integer value to String value and then finding its length and assigning the length to an integer variable:

Algorithm:

**Step-1:** Get an integer number either by initialization or by user input.

**Step-2:** Calculate total number of digits present in the input number.

**Step-3:** Keep a copy of the original number to compare it with the new number.

**Step-4:** Extract last digit from input number and raise the power of it (i.e., number of digits)

**Step-5:** Add the value to sum and decrement total number of digits value.

**Step-6:** Repeat step 4 and 5, until input number is greater than 0.

**Step-7:** After getting a final sum value compare it with the copy of the input value. If both are equal then the number is a disarium number else not a disarium number.

Multiple Approaches:

We have provided the solution in 3 different approaches.

1. By Using Static Input Value
2. By Using User Input Value
3. By Using User Defined Method

Let’s see the program along with its output one by one.

Approach-1: 12345

In this approach one integer value will be initialized in the program and then by using the algorithm we can check whether a number is a disarium number or not.

Program:

import java.util.Scanner;

public class Main

{

    //main method

    public static void main(String[] args)

    {

Output:

The original number is:89

89 is a disarium number

Approach-2: XYZ

In this approach the user will be asked to take the input of an integer value and then by using the algorithm we can check whether a number is a disarium number or not.

Program:

import java.util.Scanner;

public class Main

{

    //main method

    public static void main(String[] args)

    {

        //Created the object of Scanner class to take input

        Scanner sc = new Scanner(System.in);

        //Asking the user to enter a number

        System.out.println("Enter a number:");

        //

            //raise it to the power of its position and add it to sum

            sum = sum + (int) Math.pow(lastDigit, totalDigits);

            //Again find the original number by removing that last number

            originalNumber = originalNumber / 10;

            //reduce

        else

        {

           System.out.println(copyOfOriginalNumber + " is not a disarium number");

        }

    }

}

Output:

175 is a disarium number

Approach-3: ABCD

In this approach the user will be asked to take the input of an integer value and then we will call a user defined method by passing this input number as parameter.

Inside the method we will check whether a number is a disarium number or not by using the algorithm.

Program:

import java.util.Scanner;

public class Main

{

            //extract the last digit

            int

/ 10;

            //reduce total number of digits value by 1

            totalDigits--;

        }

        //check if sum value is equal to the copied value of original number

        //then the number is a disarium nu}

Output:

Enter a number:

175

175 is a disarium number

In this article, we explored how to check a number whether it is a disarium number or not in Java by using three different approaches.