**QUESTION:**

Create a class Disarium having following specification:

**Class Name:** Disarium

**Data Methods:**

* int n: to store the number

**Member methods:**

* MagicNumber(int nn): to initialize n with nn.
* Static void input(): to take the input number.
* int count\_digit(int): to count and return the number of digits of the given number.
* boolean check(): to check and return true if disarium number, else return false.

Implement main() to create digit, call the methods properly and print the message accordingly.

****ALGORITHM****

****Define a class called Disarium.****

****Declare an instance variable 'n' to store the input number.****

****Create a constructor 'Disarium (int nn)' to initialize 'n' with the provided value.****

****Algorithm for 'input' method:****

1. Display a message asking the user to enter a number.
2. Read an integer 'n' from the user using the 'Scanner' class.
3. Assign the value of 'n' to the static variable 'n'.

****Algorithm for 'count\_digit' method:****

1. Accept an integer 'a' as input.
2. Initialize a variable 'c' to 0 to count the number of digits.
3. Use a loop to iterate while 'a' is not equal to 0:
   * Divide 'a' by 10 and update the value of 'a' with the result.
   * Increment the value of 'c' by 1.
4. Return the value of 'c'.

****Algorithm for 'check' method:****

1. Initialize variables 'l' and 's' with the value of the input number 'n'.

****Algorithm for 'check' method:****

1. Initialize variables 'i' and 's' with the value of 1.
2. Initialize a variable 'x' with the value of 'count\_digit(n)'.
3. Use a loop to iterate while 'l' is not equal to 0:
   * Extract the last digit of 'l' and store it in 'd'.
   * Calculate the power of 'd' raised to 'x' and store it in 'i'.
   * Add 'i' to 's'.
   * Remove the last digit from 'l'.
   * Decrement the value of 'x' by 1.
4. If 's' is equal to 'n', return true.
5. Otherwise, return false.

****Create a provided constructor.****

**The provided value is initialized with 'n'.**

1. Get the number of digits in the input number n using the count\_digit method and store it in variable x.
2. Initialize a loop while l is not equal to 0:
   * Calculate the last digit of l by taking the modulo % with 10 and store it in variable d.
   * Add d raised to the power of x to the variable s.
   * Decrement the value of x by 1.
   * Divide l by 10 and update the value of l with the result.
3. If s is equal to the original input number n, return true, else return false.

**Algorithm for** main **method:**

1. Initialize a boolean variable y to false.
2. Call the input method to read the input number and assign it to the static variable n.
3. Create an instance of the Disarium class called obj, passing the value of n as a parameter to the constructor.
4. Call the check method using the obj instance and store the result in the variable y.
5. If y is true, display "The number is a Disarium number", else display "The number is not a Disarium number".

**Variable Description Table**

|  |  |  |
| --- | --- | --- |
| **Variable Name** | **Data Type** | **Description** |
| n | int | Instance variable to store the input number |
| nn | int | Parameter for the constructor |
| sc | Scanner | Scanner object for reading user input |
| c | int | Count of digits |
| a | int | Temporary variable for counting digits |
| i | int | Temporary variable for calculations |
| s | int | Sum of digits raised to power of position |
| x | int | Number of digits in the input number |
| d | int | Temporary variable for storing a digit |
| y | boolean | Result indicating if the number is Disarium |
| obj | Disarium | Instance of the Disarium class |