Algorithms and Data Structure Lab 2

The task is to take in a credit card number, and find out if it is a valid credit card number or not. Credit card numbers follow an algorithm called "Luhn's algorithm".

The formula verifies a number against its check digit, which is the last digit. This number must pass the following test:

From the rightmost digit, which is the check digit, and moving left, double the value of every second digit. If the result of this doubling operation is greater than 9 (e.g., 8 × 2 = 16), then add the digits of the product (e.g., 16: 1 + 6 = 7, 18: 1 + 8 = 9) or, alternatively, the same result can be found by subtracting 9 from the product (e.g., 16: 16 - 9 = 7, 18: 18 - 9 = 9). Take the sum of all the digits.

If the total modulo 10 is equal to 0 (if the total ends in zero) then the number is valid according to the Luhn formula; else it is not valid.

Assume an example of an account number "7992739871" that will have a check digit added, making it of the form 7992739871x. The sum of all the digits, processed as per steps 1 and 2, is 67+x. Thus x must be 3 to bring the total to be modulo 10 = 0. If x is not 3, then this is not a valid credit card number.

Input Format
An n-digit credit card number, where the last digit is the check digit.

Output Format
Output "VALID" if it is a valid credit card number and "INVALID" if it is not.

Constraints
4<=n<=30

Sample Input
4539682995824395

Sample Output
VALID

Explanation
Make sure to read from right to left.

Every second digit from the last digit.
4539682995824395
5 + 3 + 2 + 5 + 9 + 8 + 9 + 5 = 46

Double every second digit from the second last digit and subtract 9 if needed.
4539682995824395
2 * 9 = 18; 18 - 9 = 9
2 * 4 = 8
2 * 8 = 16; 16 - 9 = 7
2 * 9 = 18; 18 - 9 = 9
2 * 2 = 4
2 * 6 = 12; 12 - 9 = 3

2 * 3 = 6
2 * 4 = 8

9 + 8 + 7 + 9 + 4 + 3 + 6 + 8 = 54

Adding these together
46 + 54 = 100
100 % 10 == 0 So the card is VALID