

Formula Luhn

Tú estás desarrollando una aplicación para transacciones online y quieres aceptar muchas tarjetas de crédito. Los números de estas tarjetas, sin embargo, son usualmente largas, tanto que es fácil cometer errores cuando lo ingresas. Tú quieres crear un programa que verifique los números ingresados por el usuario.

Tú sabes que la Formula Luhn aplica para todos los números de crédito aceptables.

La fórmula Luhn funciona de la siguiente manera.

Primero, separa los dígitos individualmente de las tarjetas de crédito. Por ejemplo:

21378 se vuelve

2 1 3 7 8

Sí la cantidad de dígitos es par, multiplica por 2 cada digito que se encuentra en una posición impar. De otra manera multiplica por 2 cada digito que se encuentra en una posición par. Posiciones están indexadas desde el 1, de tal manera que el primer digito está en la posición 1. El número del ejemplo anterior contiene una cantidad de dígitos impar, entonces nosotros multiplicamos por 2 cada dígito que está en una posición par.

2 1 3 7 8 se vuelve

2 2 3 14 8

Nótese que la posición par se refiere al número original, de tal manera que ellos no cambian inclusive cuando un número de 2 dígitos aparece.

Finalmente, tomamos la suma de todos los dígitos (para la suma del número de 2 dígitos inserta ambos dígitos separadamente en la suma):

$2+2+3+1+4+8 = 20$

Si la suma es un múltiplo de 10, el número es válido, en caso contrario, es inválido.

Input

La primera línea del INPUT comienza con un número M ($1 \leq M \leq 100$), M líneas le siguen. Cada línea contiene un string P que contiene entre 0 y 50 (inclusive) caracteres representando el número de tarjeta de crédito, cada carácter será un dígito ('0'-'9').

Output

Ud. Deberá formatear su salida de la siguiente forma "Caso #N: V" (Comillas para claridad) donde N es el número de caso y V es "VALID" en caso sea válido o "INVALID" (Comillas para claridad) en caso sea un número de tarjeta no valido.

Input	Output
5	Caso #1: VALID
21378	Caso #2: INVALID
31378	Caso #3: VALID
11111101	Caso #4: VALID
50005	Caso #5: INVALID
542987223412	