ACSL  
**American Computer Science League  
Luhn Algorithm**

#### All-Star #6

**2012 - 2013**

**PROBLEM:** The Luhn Algorithm is a check sum formula used to validate identification numbers such as those used on credit cards. The last digit of most identification numbers is the check sum. The Luhn Algorithm can also be used to find several simple kinds of transcription errors. To determine if a number is valid using the check sum method use the following algorithm:

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Number | 7 | 9 | 9 | 2 | 7 | 3 | 9 | 8 | 7 | 1 | 3 |
| Doubles | 7 | 18 | 9 | 4 | 7 | 6 | 9 | 16 | 7 | 2 | x |
| Sum of Digits | 7 | 9 | 9 | 4 | 7 | 6 | 9 | 7 | 7 | 2 | 67 |

1. Counting from the check sum digit, double the value of every second digit.  
 2. If any of those doublings cause the value to be greater than 9, then change the value to  
 the sum of its digits.   
 3. Sum all the values except for the check sum.  
 4. Multiple the sum by 9  
 5. Calculate the product MOD 10.   
 6. If the calculated value is equivalent to the check sum, the number is valid.

Example #1:

67 x 9 = 603. 603 MOD 10 = 3. Since this matches the check sum digit, the number is valid.

The most common error is the single digit error - when a digit is written as an adjacent digit. That is a 4 is written as a 3 or a 5. The next most common error is a transposition error. That is, 57 is written as 75. The next most common error is twin error. That is, 77 is written as 66 or 88.

Example #2 Single Digit Error: 78927398713 is not valid. The check sum is 5. Starting with the left- most digit try adjacent digits (±1). Neither 6 nor 8 give a check sum of 3. Moving to the right again try adjacent digits. 7 gives a check sum of 7. A 9 gives a checksum of 3 and is the correct digit. Note that a 9 would become a 0 when 1 is added. A 0 would become a 9 when 1 is subtracted.

Example#3 Transposition Error: 97927398713. Start on the left and reverse 2 adjacent digits. Reversing the 97 to 79 gives the correct check sum.

Example #4 Twin Error: 78827398713. Look for twin numbers starting on the left. Use only adjacent digits (±1). Finding the 88, try 77 and then 99. The 99 produces the correct check sum.

**INPUT:** There will be 10 lines of input. Each line will contain a numerical character string (maximum length 15).

**OUTPUT:** For input lines 1 - 4. Verify the check sum digit. If it is correct print VALID. If it is not correct print the correct check sum digit. For input lines 5 and 6, the check sum digit will be correct but the number will contain a single digit error. Starting on the left, find the first number that produces the given check sum digit. Print the correct number with its check sum digit. For inputs 7 and 8 the check sum digit will be correct but the number will contain a transposition error. Starting on the left, transpose two digits and find the first number that produces the given check sum digit. Print the correct number with its check sum digit. For inputs 9 and 10, the check sum digit will be correct but the number will contain a twin error. Starting on the left change a single pair of twins to adjacent digits and find the first number that produces the given check sum digit. Print the correct number with its check sum digit.

SAMPLE INPUT SAMPLE OUTPUT

1. 235438 1. 2  
2. 198235 2. 4  
3. 210372 3. 7  
4. 987653 4. 1  
5. 112358 5. 122358  
6. 122458 6. 112458  
7. 122538 7. 122358  
8. 210737 8. 210377  
9. 366883 9. 355883  
10. 355993 10. 355883

ACSL

**American Computer Science League  
  
Luhn Alorithm**

**2012 - 2013**

#### All-Star #6

**TEST DATA**

**TEST INPUT TEST OUTPUT**

1. 2365 1. 0
2. 43256 2. 7
3. 670321 3. 3
4. 81592578 4. VALID
5. 65977 5. 66977
6. 7251 6. 7351
7. 4564789 7. 5464789
8. 11236547983 8. 11263547983
9. 32234458 9. 32235558
10. 20044661 10. 20055661