*Credit Card Validator*

### *This is a simple project that uses Luhn’s algorithm to validate a user's credit card. The program works for all popular cards like Visa, Amex, MasterCard, etc. Luhn’s algorithm checks for basic validations; for example, a Visa card should start with 4 and then moves on to complex digit-wise calculations. It is a good program to learn because most e-commerce transactions require credit card validation.*

# *Luhn algorithm*

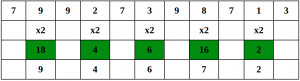
*The Luhn algorithm, also known as the****modulus 10****or****mod 10****algorithm, is a simple checksum formula used to validate a variety of identification numbers, such as credit card numbers, IMEI numbers, Canadian Social Insurance Numbers. The LUHN formula was created in the late 1960s by a group of mathematicians. Shortly thereafter, credit card companies adopted it. Because the algorithm is in the public domain, it can be used by anyone. Most credit cards and many government identification numbers use the algorithm as a simple method of distinguishing valid numbers from mistyped or otherwise incorrect numbers. It was designed to protect against accidental errors, not malicious attacks.*

#### Steps involved in the Luhn algorithm

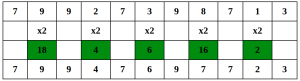
Let’s understand the algorithm with an example:   
Consider the example of an account number “**79927398713**“.   
**Step 1**– Starting from the rightmost digit, double the value of every second digit,



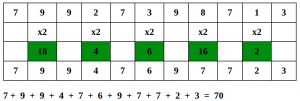
**Step 2** – If doubling of a number results in a two digit number i.e greater than 9(e.g., 6 × 2 = 12), then add the digits of the product (e.g., 12: 1 + 2 = 3, 15: 1 + 5 = 6), to get a single digit number.



**Step 3**– Now take the sum of all the digits.



**Step 4**– 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.



**Since the sum is 70 which is a multiple of 10, the account number is possibly valid.**