pdp 2.ödevi luhn algoritması

Ve make file

Sakarya üniversitesi bil. Sis. Müh. bölümü

B201200571 başşar şaraf

# Difinetion of luhn algorithm: (www.geeksforgeeks.org, 2022)

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

An example of luhn algorithm :

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.

The idea is simple; we traverse from the end. For every second digit, we double it before adding it. We add two digits of the number obtained after doubling.

# Advantages and Disadvantages of luhn algorithm:

The Luhn algorithm will detect any single-digit error, as well as almost all transpositions of adjacent digits. It will not, however, detect transposition of the two-digit sequence *09* to *90* (or vice versa). It will detect most of the possible twin errors (it will not detect *22* ↔ *55*, *33* ↔ *66* or *44* ↔ *77*).

Other, more complex check-digit algorithms (such as the Verhoeff algorithm and the Damm algorithm) can detect more transcription errors. The Luhn mod N algorithm is an extension that supports non-numerical strings.

Because the algorithm operates on the digits in a right-to-left manner and zero digits affect the result only if they cause shift in position, zero-padding the beginning of a string of numbers does not affect the calculation. Therefore, systems that pad to a specific number of digits (by converting 1234 to 0001234 for instance) can perform Luhn validation before or after the padding and achieve the same result.

The algorithm appeared in a United States Patent[[1]](https://en.wikipedia.org/wiki/Luhn_algorithm#cite_note-patent-1) for a hand-held, mechanical device for computing the checksum. Therefore, it was required to be rather simple. The device took the mod 10 sum by mechanical means. The *substitution digits*, that is, the results of the double and reduce procedure, were not produced mechanically. Rather, the digits were marked in their permuted order on the body of the machine. (en.wikipedia.org, 2022)

# What are the Usages of Luhn Algorithm?

The Luhn Algorithm—also known as the “Modulus 10 Algorithm”—is a formula that is used to determine whether the identification number provided by a user is accurate. The formula is widely used in validating credit card numbers, as well as other number sequences such as government Social Security Numbers

## Real-World Example of the Luhn Algorithm’s usages:

One of the central concepts within the Luhn Algorithm is the use of so-called “check digits.” These digits consist of numbers that are inserted into the broader number sequence in order to help verify, or “check,” whether the whole number is authentic.

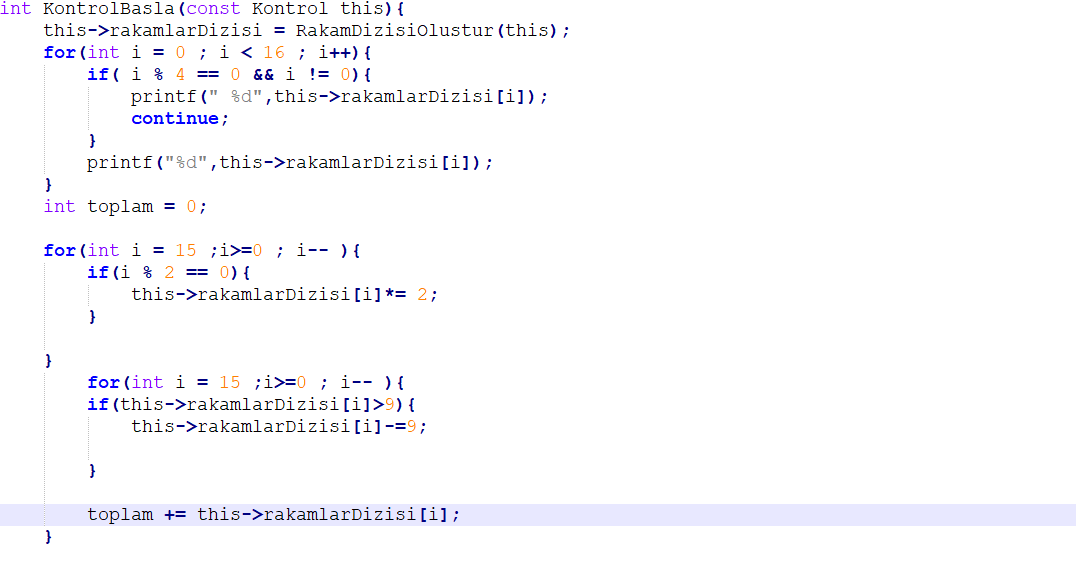
For credit cards, the check digit consists of a single digit printed at the end of the credit card number. Rather than being specifically chosen by the credit card company, the check digit is instead automatically determined by the Luhn Algorithm, based on the preceding numbers in the sequence. When users enter their credit card numbers to complete transactions, the payment processing software can use the Luhn Algorithm to detect whether the specified number is accurate, based in part on its check digit.

Today, the Luhn Algorithm is integrated into popular programming languages and code libraries, making it relatively easy to include Luhn-based identification number verification in new software applications. (www.investopedia.com, 2022)

# How I did the Project :

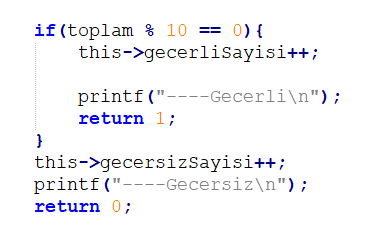
At first I created a library to interact with text files and it contains functions to read,write and calculate the number of lines in a text file

Then I started the luhn algorithm in “kontrol.c” file I started from the most right digit and multiplied the second forth .... digits by 2 and by this I get a new sequince of numbers



Then I collected the digits of the number which contain more than one digit “two digits” but i did it a different way by subtract 9 from the number and this is the third step

Number’s which are divisible by ten are valid numbers



İt is important to mention that I used a trick to use structs like a class and scince structs can not contain function I made a variable instead of every method and it’s value is a pointer refers to the methods i wrote

finally “yikici metot ” will free the memory used for variables i used.

