

VALIDADOR E GERADOR DE CÓDIGO DE BARRAS (PADRÃO EAN-13)

Contexto

O código de barras EAN-13 é um padrão internacional composto por 13 dígitos. Os primeiros 12 dígitos representam informações sobre o produto (país de origem, fabricante e código do produto), enquanto o 13º dígito é um **dígito de controle**, calculado matematicamente a partir dos 12 anteriores. A função desse dígito é garantir a integridade do código, ajudando a detectar erros de leitura ou digitação.

O Desafio

Seu objetivo é criar um programa que possa executar duas funções principais:

1. **Validar um código EAN-13 completo:** Receber um código de 13 dígitos e determinar se ele é válido, verificando se o dígito de controle está correto.
2. **Gerar um código EAN-13:** Receber os 12 primeiros dígitos e calcular o 13º dígito de controle correto para formar um código completo e válido.

As Regras do Jogo: O Cálculo do Dígito de Controle

Para validar ou gerar o dígito de controle, você deve usar os **12 primeiros dígitos** do código e seguir estritamente o algoritmo abaixo:

1. Começando da **esquerda para a direita**, some todos os dígitos que estão em posições **ímpares** (o 1º, 3º, 5º, 7º, 9º e 11º dígito).
2. Some todos os dígitos que estão em posições **pares** (o 2º, 4º, 6º, 8º, 10º e 12º dígito).
3. Multiplique o resultado da soma dos dígitos pares por **3**.
4. Some os resultados dos passos 1 e 3 para obter um total.
5. O dígito de controle (o 13º dígito) é o **menor número** que, quando somado a esse total, resulta em um número que é **múltiplo de 10**.

Observação: Se a "soma total" do cálculo já for um múltiplo de 10, o dígito de controle é **0**.

Requisitos do Algoritmo

Você deve criar duas funções (ou métodos):

1. validar(codigo_completo)

- **Entrada:** Uma string de 13 caracteres numéricos.

- **Saída:** Um valor booleano (true se o código for válido, false caso contrário).

- Testes:

- validar("7891000315477") deve retornar true.

- validar("7891000315478") deve retornar false.

- validar("4003994155486") deve retornar true.

2. gerar(codigo_parcial)

- **Entrada:** Uma string de 12 caracteres numéricos.

- **Saída:** Uma string de 13 caracteres, correspondente ao código completo com o dígito de controle correto.

- Testes:

- gerar("789100031547") deve retornar "7891000315477".

- gerar("400399415548") deve retornar "4003994155486".

- gerar("978020137962") deve retornar "9780201379624".

PROGRAMMING CHALLENGE : EAN-13 BARCODE VALIDATOR AND GENERATOR

Background

The EAN-13 barcode is an international standard composed of 13 digits¹. The first 12 digits represent information about the product (country of origin, manufacturer, and product code), while the 13th digit is a check digit, mathematically calculated from the previous 12². The purpose of this digit is to ensure the code's integrity, helping to detect reading or typing errors³.

The Challenge

Your objective is to create a program that can perform two main functions⁴:

1. **Validate a complete EAN-13 code:** Receive a 13-digit code and determine if it is valid by verifying that the check digit is correct⁵.
 2. **Generate an EAN-13 code:** Receive the first 12 digits and calculate the correct 13th check digit to form a complete and valid code⁶.
-

The Rules: Calculating the Check Digit

To validate or generate the check digit, you must use the first 12 digits of the code and strictly follow the algorithm below⁷:

1. Starting from left to right, sum all digits that are in **odd-numbered positions** (the 1st, 3rd, 5th, 7th, 9th, and 11th digits)⁸.
2. Sum all digits that are in **even-numbered positions** (the 2nd, 4th, 6th, 8th, 10th, and 12th digits)⁹.
3. Multiply the result from the sum of the even-numbered digits by **3**¹⁰.
4. Add the results from steps 1 and 3 to get a **grand total**¹¹.
5. The **check digit** (the 13th digit) is the smallest number that, when added to the grand total, results in a number that is a **multiple of 10**¹².

Note: If the grand total is already a multiple of 10, the check digit is 0¹³.

Algorithm Requirements

You must create two functions (or methods)¹⁴:

1.

`validate(full_code)` ¹⁵

- **Input:** A string of 13 numeric characters¹⁶.
- **Output:** A boolean value (true if the code is valid, false otherwise)¹⁷.
- **Tests:**
 - `validate("7891000315477")` should return true¹⁸.
 - `validate("7891000315478")` should return false¹⁹.
 - `validate("4003994155486")` should return true²⁰.

2.

`generate(partial_code)` ²¹

- **Input:** A string of 12 numeric characters²².
- **Output:** A 13-character string, corresponding to the complete code with the correct check digit²³.
- **Tests:**
 - `generate("789100031547")` should return "7891000315477"²⁴.
 - `generate("400399415548")` should return "4003994155486"²⁵.
 - `generate("978020137962")` should return "9780201379624"²⁶.