

**Universidade do Estado do Rio de Janeiro**  
**Instituto de Matemática e Estatística**  
**Departamento de Informática e Ciência da Computação**



**ESTUDO E IMPLEMENTAÇÃO DE DÍGITOS  
VERIFICADORES**

Autores: **DANIEL ARGENTA**  
**E RAFAEL AMORIM**

RIO DE JANEIRO  
Fevereiro/2012

## **RESUMO**

Os dígitos verificadores são fundamentais no universo tecnológico. Estão presentes em incontáveis informações que são armazenadas e enviadas, especialmente no contexto comercial. Sem esses códigos, o número de informações, principalmente na entrada de dados, seria muito pouco confiável, em decorrência especialmente dos erros humanos na digitação. Devido a grande importância dos dígitos verificadores, esse é o tema escolhido para a monografia de fim da graduação em Informática e Tecnologia de Informação da Universidade do Estado do Rio de Janeiro.

O objetivo principal deste trabalho é fazer uma apresentação didática dos esquemas de dígitos verificadores utilizados nos mais variados contextos. Apresentaremos, também, os fundamentos matemáticos que levaram à existência dessa tecnologia. Nos capítulos finais, particularizamos para a situação brasileira, e fazemos uma análise dos esquemas usados no Brasil, bem como a implementação de um grande número desses esquemas.

# SUMÁRIO

<b>CAPÍTULO 1 – INTRODUÇÃO .....</b>	<b>5</b>
1.1 DÍGITOS VERIFICADORES .....	5
1.2 OBJETIVOS DO PROJETO.....	6
1.3 ORGANIZAÇÃO DA MONOGRAFIA .....	6
<b>CAPÍTULO 2 – IDENTIFICADORES E DÍGITOS VERIFICADORES.....</b>	<b>8</b>
2.1 TIPOS DE CÓDIGOS.....	8
2.2 ERROS DE DIGITAÇÃO, TRANSCRIÇÃO E TRANSMISSÃO .....	8
2.3 DETECÇÃO E CORREÇÃO DE ERROS .....	9
2.3.1 <i>Esquemas de detecção de erros</i> .....	9
2.3.2 <i>Esquemas de correção de erros</i> .....	10
2.4 DÍGITOS VERIFICADORES .....	12
<b>CAPÍTULO 3 – A BASE MATEMÁTICA DOS DÍGITOS VERIFICADORES.....</b>	<b>13</b>
3.1 INTRODUÇÃO .....	13
3.2 DIVISÃO DE INTEIROS .....	13
3.3 ARITMÉTICA MODULAR.....	14
3.4 MAPEAMENTO ANTI-SIMÉTRICO .....	14
3.4.1 <i>Conceitos da Teoria dos Grupos aplicados aos Dígitos Verificadores</i> .....	15
3.4.2 <i>Mapeamento antissimétrico</i> .....	16
3.4.3 <i>O Esquema de Verhoeff</i> .....	17
<b>CAPÍTULO 4 – ESQUEMAS DE DÍGITOS VERIFICADORES DIFUNDIDOS NO BRASIL .....</b>	<b>19</b>
4.1 IDENTIFICAÇÃO POR CÓDIGO DE BARRAS .....	19
4.2 IDENTIFICAÇÃO DO CADASTRO DE PESSOA FÍSICA – CPF .....	21
4.3 IDENTIFICAÇÃO DA CARTEIRA NACIONAL DE HABILITAÇÃO – CNH .....	22
4.4 IDENTIFICAÇÃO DO PASSAPORTE DA REPÚBLICA FEDERATIVA DO BRASIL.....	23
4.5 IDENTIFICAÇÃO DE REGISTRO GERAL .....	25
4.6 IDENTIFICAÇÃO DO TÍTULO ELEITOR .....	27
4.7 IDENTIFICAÇÃO DO PIS / PASEP .....	28
4.8 IDENTIFICAÇÃO DA CARTEIRA DE TRABALHO.....	29
4.9 IDENTIFICAÇÃO DO CERTIFICADO DE RESERVISTA .....	29
4.10 IDENTIFICAÇÃO DO CARTÃO DE CRÉDITO .....	29
4.11 IDENTIFICAÇÃO DO BLOQUETO DE COBRANÇA .....	30
4.12 IDENTIFICAÇÃO DE CÓDIGO DE BARRAS DE TRIBUTOS/TAXAS ESTADUAIS E MUNICIPAIS E CONTAS DE CONCESSIONÁRIAS DE SERVIÇOS PÚBLICOS. .....	32
4.13 IDENTIFICAÇÃO DO CADASTRO NACIONAL DE PESSOA JURÍDICA - CNPJ .....	36
4.14 IDENTIFICAÇÃO DE AGÊNCIA E CONTA BANCÁRIA – BANCO DO BRASIL.....	38
4.15 IDENTIFICAÇÃO DE AGÊNCIA E CONTA BANCÁRIA – BRADESCO .....	39
4.16 IDENTIFICAÇÃO DE AGÊNCIA E CONTA BANCÁRIA – CAIXA ECONÔMICA FEDERAL .....	40
<b>CAPÍTULO 5 – IMPLEMENTAÇÃO DE SITE.....</b>	<b>41</b>
<b>CAPÍTULO 6 - CONCLUSÃO .....</b>	<b>50</b>
<b>REFERÊNCIAS .....</b>	<b>51</b>

## **LISTA DE TABELAS**

- Tabela 1. Principais tipos de erros na digitação de dados
- Tabela 2. Exemplo de código Hamming
- Tabela 3. Condições pra detecção dos principais tipos de erros
- Tabela 4. Tabela Cayley para  $D_{10}$  de Verhoeff
- Tabela 5. Cálculo código de barras EAN
- Tabela 6. Unidades federativas do CPF
- Tabela 7. Unidade da federação do título de eleitor
- Tabela 8. Campos do código de barras do bloqueto de cobrança
- Tabela 9. Campos do identificador do bloqueto de cobrança
- Tabela 10. Exemplo do identificador do bloqueto de cobrança
- Tabela 11. Campos do identificador do código de barras de serviços públicos

## **LISTA DE FIGURAS**

## **LISTA DE TERMOS**

# Capítulo 1 – Introdução

Este tópico está fortemente baseado nos artigos descritos em [1] e [2].

## 1.1 Dígitos verificadores

Dígitos verificadores são mecanismos que utilizam um ou mais dígitos (numéricos ou alfanuméricos), acrescentados a uma cadeia de dígitos (numérica ou alfanumérica) original, que certifica e/ou corrige esta cadeia, dando maior segurança contra fraudes, erros de digitação ou leitura (através de um scanner, por exemplo). Esses dígitos são formulados através de algoritmos que podem ser públicos ou não.

Tais dígitos são utilizados em códigos de barra de produtos, números de conta corrente, documentos de identidade, CPF,CNPJ, etc. com as finalidades mencionadas anteriormente e têm grande importância no mundo comercial, na identificação civil, arrecadação de tributos, instituições financeiras e muitas outras áreas, onde seja vantajoso.

A principal proteção dada pelos dígitos verificadores é na entrada de dados, onde é comum haver a transcrição errada de documentos importantes, principalmente por erros humanos, que podem ter sua freqüência reduzida. Entretanto, sempre estarão presentes em maior ou menor grau.

Existem diversos tipos de erros. Os mais comuns foram categorizados por J. Verhoeff [3], baseado em um estudo de 12.000 erros. A Tabela 1 apresenta os principais erros humanos e suas freqüências, medidas empiricamente.

Esse estudo foi elaborado considerando a língua inglesa, mas é razoável supor que , excluindo-se os erros fonéticos do idioma anglo-saxão , podemos adotá-lo para outras culturas em outras linguagens. Não foi encontrado um estudo de erros fonéticos equivalente ao de Verhoeff para os idiomas latinos.

Tipo de Erro	Formato	Frequencia de Ocorrência
Erros individualizados	....a....→....b....	79,10%
Transposição de Digitos Adjacentes	...ab...→...ba...	10,20%
Transposição Alternada	...abc...→...cba...	0,80%
Erros Gêmeos	...aa...→...bb...	0,50%
Erros Fonéticos	a0↔1a	0,50%
	a=2,...9	
Erros Gêmeos Alternados	aca→bcb	0,30%

Tabela 1. Principais tipos de erros na digitação de dados

Para detectar os erros da Tabela 1, normalmente adiciona-se uma redundância aos códigos. Esta redundância consta de um ou mais dígitos verificadores e, na maioria das

vezes, permite apenas sinalizar a ocorrência de erro. Em alguns casos, os dígitos acrescentados podem inclusive permitir a correção do erro, mas são mais difíceis e raros de serem utilizados.

A adição de dígitos verificadores é baseada em alguns esquemas matemáticos que são pouco difundidos, de forma que, quando estamos diante de determinados esquemas de detecção ou correção de erros, podemos não saber exatamente quais as possibilidades de detecção ou correção que o mecanismo permite, uma vez que são divulgados apenas os algoritmos de checagem, e não a base matemática de sua construção.

Dada a importância e onipresença no nosso quotidiano desses esquemas, é importante conhecê-los melhor.

## 1.2 Objetivos do Projeto

Como foi visto, os dígitos verificadores desempenham um papel importante no nosso dia a dia e são pouco conhecidos. Em particular, existe no Brasil uma série de esquemas específicos para a detecção de erros.

Este projeto tem como objetivo sanar algumas das omissões apontadas:

- a) Apresentar as principais bases matemáticas utilizadas nos esquemas de dígitos verificadores.
- b) Descrever e implementar os algoritmos envolvidos, disponibilizando-os em uma página Web criada para esses fins e vinculada às páginas do IME/UERJ.
- c) Apresentar uma análise de esquemas de detecção de erros utilizadas especificamente no Brasil, tais como o CPF, CNH, Identidade etc.

## 1.3 Organização da monografia

O Capítulo 2 introduz alguns conceitos relacionados a identificadores e dígitos verificadores, necessários para o restante do texto

No capítulo 3 mostraremos alguns esquemas de dígitos verificadores utilizados mundialmente e descreveremos a Matemática subjacente aos mesmos

No Capítulo 4 apresentamos inúmeros esquemas utilizados no Brasil. Fazemos uma análise da efetividade desses esquemas. Apresentamos, também uma pequena ferramenta didática, para ilustrar os esquemas estudados. Tentamos trazer uma abordagem relacionada mais aos algoritmos do que em relação a teoria matemática , muito diferente de quase todas as fontes de pesquisa que foram utilizadas. Não mostraremos os códigos dos programas, mas descreveremos as telas e faremos uma breve modelagem do sistema.

No Capítulo 5, descrevemos uma página da Web contendo as demonstrações dos principais esquemas de dígitos verificadores utilizados no Brasil. Mostramos e descrevemos a interface utilizada nessa ferramenta.

No Capítulo 6, de conclusão fazemos um resumo do trabalho realizado e encorajaremos novos estudantes para continuar este trabalho, acrescentando na pagina web outros códigos verificadores e corretores conhecidos.

# Capítulo 2 – Identificadores e Dígitos Verificadores

Este capítulo está fortemente baseado no artigo descrito em [4].

Os Identificadores são utilizados para possibilitar a referência individualizada a produtos, documentos e contas, sendo também importantes para o rastreamento de entregas, logística de estoques etc. A esses identificadores normalmente são acrescentados dígitos verificadores.

## Tipos de Códigos

Usualmente, um identificador tem o formato de um string de dígitos (0, 1,...,9), letras, símbolos (\*, %, \$, # ...) ou uma combinação deles.

Para a maioria dos casos, duas informações a respeito do numero identificador são primordiais:

- O comprimento, ou quantidade de caracteres.
- A posição de cada caractere.

Utilizaremos a seguinte convenção para descrever um código (identificador):  $a_1a_2a_3...a_n$ , onde  $n$  denota o comprimento do identificador, ou seja, a quantidade total de dígitos, letras e símbolos contidos nele e o índice indica a posição de cada caractere onde  $a_1$  é o primeiro caractere,  $a_2$  o segundo,  $a_3$  o terceiro, até  $a_n$  que é o último caractere.

## 2.2 Erros de digitação, transcrição e transmissão

Identificadores aparecem constantemente em conversas telefônicas, documentos, digitações e digitalizações, enviados através da internet ou transmitidos de alguma outra forma. A cada vez que isso acontece, há a possibilidade de que um ou mais dígitos sejam alterados ou deslocados, quando são movidos. É importante garantir que os identificadores sejam transmitidos de forma correta. As formas mais comuns de erro são as que envolvem o manuseio das informações por parte do homem, o que geralmente ocorre na hora de digitar as informações.

Conforme mencionado anteriormente, os principais tipos de erros são: *erro-individualizado*, *transposição de dígitos adjacentes*, *transposição alternada*, *erro gêmeo, fonético e gêmeo alternado*.

O receptor do número, não tem como identificar quando o número está correto, a não ser que o remetente possa ser consultado. Nem sempre esse contato é possível ou viável. Isso motivou a criação de um método em que o receptor possa perceber quando um número identificador foi transmitido de forma incorreta. Ele é chamado de Método de Dígito Verificador. Existem algoritmos bastante difundidos, sendo que alguns inclusive

identificam e corrigem os erros automaticamente, chamados “Códigos corretores de erros”.

## 2.3 Detecção e correção de erros

Este tópico está fortemente baseado nos artigos descritos em [5] e [6].

Os princípios matemáticos para detecção e correção de erros utilizam uma métrica básica denominada "*Distância Hamming*".

Dados dois identificadores  $x$  e  $y$ , define-se a *distância Hamming* entre eles  $d(x, y)$  como o total de posições em que eles diferem. Por exemplo, a distância Hamming entre '245a' e '254a' é de 2, já que diferem nas posições 2 e 3. Denominamos *código* o conjunto de identificadores construídos com um conjunto básico de caracteres. Dado um código  $C$ , definimos também a *distância mínima* do código, denotada por  $d(C)$  como a menor distância Hamming entre os identificadores do código. O teorema a seguir, demonstrado em [7], é um dos princípios básicos para se adotar detecção ou correção de erros, nos esquemas de entrada ou transmissão de dados.

**Teorema:** Dado um Código  $C$ , é possível detectar-se  $s$  erros se o código tem distância mínima  $d(c) \geq s+1$  e é possível corrigir-se  $t$  erros se temos  $d(C) \geq 2t+1$ .

Este teorema quantifica o princípio intuitivo de que o preço a pagar para corrigir erros é sempre maior do que aquele usado para detectar erros apenas. Devido a isso, é muito mais comum haver esquemas de detecção do que de correção de erros.

A aplicação de códigos detectores e corretores de erros não é nenhuma novidade, e consiste basicamente na adição de dados redundantes criados a partir dos dados originais, a um conjunto de bits. O objetivo é fazer a operação inversa em cima dos dados redundantes e obter a informação original novamente.

### 2.3.1 Esquemas de detecção de erros

Há quatro esquemas principais de detecção de erros:

-Esquemas de repetição: A mensagem é repetida, uma ou mais vezes, e vê-se no destino quais os números mais freqüentes em cada uma das posições. Por exemplo, seja a mensagem UERJ a mensagem original. Repetindo-a 3 vezes e enviando ao destinatário, foi recebida como UERD USRJ UERJ. Como a 1<sup>a</sup> e a 3<sup>a</sup> posição estão iguais, ele as considera certas, a 2<sup>a</sup> tem a letra E com maior freqüência e a 4<sup>a</sup> a letra J como maior freqüência.

-Esquemas de paridade: O código binário é dividido em cadeias de strings de mesmo tamanho e o número de ocorrências do dígito “1” é calculado. É adicionado a este bloco um bit de paridade que pode ser “1” se este número for ímpar e “0” caso seja

par. O receptor também calcula o dígito de paridade e compara com o que foi enviado para verificar a integridade dos dados.

Por exemplo, caso o string transmitido fosse 10100010111, o receptor iria considerar o ultimo bit, neste caso “1”, como o bit de paridade e entendendo que o número de ocorrências do bit “1” é ímpar (5 ocorrências) validando a recepção.

-Checksum: Aplica operações aritméticas sobre os dígitos, normalmente a adição, e insere o resultado como dado redundante, para conferência da operação no destino. O objetivo é que as operações aritméticas sejam complexas o suficiente para que a informação redundante varie para todos os erros de transmissão, e assim poder identificar o erro.

-Redundância cíclica: considera cada dígito do código binário como sendo o coeficiente de uma equação polinomial (Ex:  $10000111 = x^8+x^3+x^2+x$ ), depois divide-se este polinômio por uma representação binária de um polinômio de menor grau pré estabelecido, sendo este resultado o dado redundante para checagem no destino. Este esquema é relevante pois o dado redundante é sempre menor que o original e quanto maior o grau do polinômio que divide o original, maior a eficácia.

### 2.3.2 Esquemas de correção de erros

Há inúmeros esquemas de correção de erros. Um dos mais famosos é o de **Hamming Codes**, descrito a seguir:

Este código parte de uma codificação original e insere bits de redundância dentro do bloco de código, nas posições  $2^n$  ( $n=1,2,4,8,16\dots$ ), deslocando os bits do código original mas mantendo a sequência da codificação original.

Por exemplo, seja **10111001** um bloco original de 8 dígitos. Este bloco será transformado para outro denominado H(12,8), nomenclatura que indica existirem 12 bits no bloco total e que pretende-se transmitir 8 bits com a utilização de 4 bits para a detecção de erros neste bloco de 8 bits a enviar.

Serão então criadas / definidas as posições **X** de redundância transformando o bloco original em **XX1X011X1001**. Na sequência, faremos as operação lógica XOR (eXclusive OR) entre cada uma das representações binárias das posições ocupadas pelo bit “1” com o resultado da operação anterior, sendo que a primeira operação será efetuada entre as duas primeiras representações binárias ocupadas pelo bit ‘1’ por não haver ainda nenhum resultado

Sendo as representações binárias das posições ocupadas pelo bit “1” no bloco total:

3 =00011 (a)

6 =00110 (b)

7 =00111 (c)

$$9 = 01001 \quad (\text{d})$$

$$12 = 01100 \quad (\text{e})$$

Aplicando o operador lógico XOR (Retorna true se um e somente um dos valores for verdadeiro(1)):

$$\begin{array}{ll} 0011 \text{ XOR } 0110 = 0101 & (\text{f}) \\ 0101 \text{ XOR } 0111 = 0010 & (\text{g}) \\ 0010 \text{ XOR } 1001 = 1011 & (\text{h}) \end{array}$$

$$1011 \text{ XOR } 1100 = \mathbf{0111} \quad (\text{i}) \quad (\text{h} \wedge \text{e})$$

Agora, invertendo (esquerda para a direita) o ultimo resultado (m) teremos os dígitos para as posições de redundância da 1<sup>a</sup> posição à 4<sup>a</sup> posição, deixando o código total conforme a Tabela 2.

X	X	1	X	0	1	1	X	1	0	0	1
1	1	1	1	0	1	1	0	1	0	0	1

Tabela 2. Exemplo de código Hamming

O código Hamming consegue corrigir erros de 1bit. Os bits recebidos são representados como  $y_1y_2y_3y_4y_5\dots y_{17}$ . Para sabermos se o código contem algum erro, devemos fazer um cálculo de 4 bits chamados  $K_1, K_2, K_3$  e  $K_4$ :

$$K_1 = y_1 \text{ xor } y_3 \text{ xor } y_5 \text{ xor } y_7 \text{ xor } y_9 \text{ xor } y_{11}$$

$$K_2 = y_2 \text{ xor } y_3 \text{ xor } y_6 \text{ xor } y_7 \text{ xor } y_{10} \text{ xor } y_{11}$$

$$K_3 = y_4 \text{ xor } y_5 \text{ xor } y_6 \text{ xor } y_7 \text{ xor } y_{12}$$

$$K_4 = y_8 \text{ xor } y_9 \text{ xor } y_{10} \text{ xor } y_{11} \text{ xor } y_{12}$$

Se  $K_1=K_2=K_3=K_4=0$ , então não ocorreu um erro na transmissão.

Senão, o número binário representado por  $K_4 K_3 K_2 K_1$  indicará a posição em que o erro ocorreu.

Para exemplificar, suponhamos que não sabemos o código original e recebemos o código 111101001001, que nada mais é que o código da Tabela 2, com um erro na posição 7( $y_7$ ). Sendo  $y_1=1, y_2=1, y_3=1, y_4=1, y_5=0, y_6=1, y_7=0, y_8=0, y_9=1, y_{10}=0, y_{11}=0$  e  $y_{12}=1$ .

$$K_1 = 1 \text{ xor } 1 \text{ xor } y_5 \text{ xor } y_7 \text{ xor } y_9 \text{ xor } y_{11}$$

$$K_2 = 0 \text{ xor } 0 \text{ xor } y_7 \text{ xor } y_9 \text{ xor } y_{11}$$

$$K_3 = 0 \text{ xor } 0 \text{ xor } y_9 \text{ xor } y_{11}$$

$$K_1 = 0 \text{ xor } 1 \text{ xor } y_{11}$$

$$K_1 = 1 \text{ xor } 0 = 1$$

$$K_2 = 1 \text{ xor } 1 \text{ xor } y_6 \text{ xor } y_7 \text{ xor } y_{10} \text{ xor } y_{11}$$

$$K_2 = 0 \text{ xor } 1 \text{ xor } y_7 \text{ xor } y_{10} \text{ xor } y_{11}$$

$$K_2 = 1 \text{ xor } 0 \text{ xor } y_{10} \text{ xor } y_{11}$$

$$K_2 = 1 \text{ xor } 0 \text{ xor } y_{11}$$

$$K_2 = 1 \text{ xor } 0 = 1$$

$$K_3 = 1 \text{ xor } 0 \text{ xor } y_6 \text{ xor } y_7 \text{ xor } y_{12}$$

$$K_3 = 1 \text{ xor } 1 \text{ xor } y_7 \text{ xor } y_{12}$$

$$K_3 = 0 \text{ xor } 0 \text{ xor } y_{12}$$

$$K_3 = 0 \text{ xor } 1 = 1$$

$$K_4 = 0 \text{ xor } 1 \text{ xor } y_{10} \text{ xor } y_{11} \text{ xor } y_{12}$$

$$K_4 = 1 \text{ xor } 0 \text{ xor } y_{11} \text{ xor } y_{12}$$

$$K_4 = 1 \text{ xor } 0 \text{ xor } y_{12}$$

$$K_4 = 1 \text{ xor } 1 = 0$$

$$K_4 K_3 K_2 K_1 = 0111 = 7$$

Ou seja,  $y_7$  sofreu alteração. Como ele foi recebido como 0, antes do erro de transmissão era 1.

## 2.4 Dígitos verificadores

A maioria dos “Métodos de dígitos verificadores” utilizados atualmente em representações numéricas que exijam integridade, insere no identificador um dígito extra, chamado dígito verificador, que é usado para identificar erros após a transmissão do mesmo. Este dígito poderia aparecer em qualquer posição ( $n$ ) do identificador, mas o mais freqüente é aparecer na última posição ( $a_n$ ).

Sistemas de verificação de integridade dos dados utilizam diferentes métodos de aplicação e cálculo dos dígitos verificadores. Alguns são mais efetivos do que outros, pois apesar de 90% dos erros serem erros individualizados ou transposição de dígitos adjacentes, qualquer método criado atualmente deveria identificar pelo menos estes tipos de erro. Entretanto, alguns dos esquemas mais simples utilizados atualmente não chegam às vezes a detectar eficazmente esses erros.

# Capítulo 3 – A base matemática dos dígitos Verificadores

Neste capítulo descreveremos de forma sucinta os princípios matemáticos utilizados nos esquemas de dígitos verificadores.

## 3.1 Introdução

Existem diversos conjuntos numéricos, que podem ser considerados. Diferentes situações requerem diferentes tipos de números. Basicamente há seis conjuntos numéricos, mas citaremos apenas cinco. São eles:

- Números Naturais não nulos,  $N^* = \{1,2,3,4,\dots\}$ ;
- Números Naturais,  $N = \{0,1,2,3,4,\dots\}$ ;
- Números Inteiros,  $Z = \{\dots,-4,-3,-2,-1,0,1,2,3,4,\dots\}$ ;
- Números Racionais,  $Q = \{\text{conjunto dos números que podem ser expressos como uma fração } p/q, \text{ onde } p \in Z \text{ e } q \neq 0\}$ ;
- Números Reais,  $R = \{\text{conjunto de números correspondentes aos pontos da reta}\}$ .

A maior parte do nosso estudo envolve os números Inteiros.

Utilizaremos em nosso estudo, as quatro operações aritméticas básicas: adição, subtração, multiplicação e divisão, que junto aos números Naturais e Inteiros formarão a base para nosso estudo a respeito dos métodos de dígitos verificadores. Recapitularemos a seguir alguns conceitos da divisão.

## 3.2 Divisão de Inteiros

Este tópico está fortemente baseado no artigo descrito em [4].

**Definição 3.2.1.** Um número inteiro  $y$  divide outro número inteiro  $x$  se há outro número inteiro  $n$ , tal que  $x = n.y$ . Denotado como  $y | x$ . Se  $y$  não divide  $x$ , então denota-se  $y \nmid x$ .

**Definição 3.2.2.** Um número inteiro é considerado primo, se este é maior do que 1 e os únicos naturais não-nulos ( $Z^*$ ) que o divide são 1 e ele mesmo. Um número inteiro maior do que 1 não-primo é chamado composto.

**Proposição 3.2.3.** Todo inteiro  $n > 1$  pode ser representado como um produto único de números primos  $n = p_1^{r_1} p_2^{r_2} \dots p_t^{r_t}$ , aonde  $p_1, p_2, \dots, p_t$  são primos distintos, sendo  $p_1 < p_2 < \dots < p_t$ , e  $r_i$ , para  $1 \leq i \leq t$  sendo a potência a qual cada primo é elevado. O produto  $p_1^{r_1} p_2^{r_2} \dots p_t^{r_t}$  é chamado de decomposição de fatores primos de  $n$ .

**Definição 3.2.4.** Dois números inteiros  $x$  e  $y$  são relativamente primos se não há um número inteiro maior do que 1 que divida a ambos.

**Teorema 3.2.5 (Algoritmo de Divisão).** Para quaisquer dois números inteiros  $x$  e  $y$ , onde  $y > 0$ , existem números inteiros únicos  $q$  e  $r$ , tais que:  $x = q \cdot y + r$  onde  $0 \leq r \leq y-1$

### 3.3 Aritmética Modular

A aritmética é o estudo das operações básicas: adição, subtração, multiplicação e divisão. A aritmética modular é o estudo das operações básicas sobre um contexto diferente, referente ao sistema de números inteiros módulo  $n$ .

**Notação:** Considerando dois números inteiros  $x$  e  $n$ , sendo  $n > 0$ . O resto  $r$  obtido quando  $x$  é dividido por  $n$  é denotado por  $x \pmod{n}$ .

Dois números podem ser somados, subtraídos ou multiplicados modulo  $n$ :

- Soma:  $(a+b) \pmod{n} = ((a \pmod{n}) + (b \pmod{n})) \pmod{n}$
- Subtração:  $(a-b) \pmod{n} = ((a \pmod{n}) - (b \pmod{n})) \pmod{n}$
- Multiplicação:  $(a \cdot b) \pmod{n} = ((a \pmod{n}) \cdot (b \pmod{n})) \pmod{n}$

**Definição 3.3.1.** Sendo  $n$  um número natural não-nulo. Dados dois números inteiros  $x_1$  e  $x_2$ ,  $x_1$  é congruente a  $x_2$  módulo  $n$ , escrito como  $x_1 \equiv x_2 \pmod{n}$ , se  $x_1$  e  $x_2$  possuírem o mesmo resto quando divididos por  $n$ .

**Observação:** O sinais de igualdade “=” e equivalência “ $\equiv$ ” são diferentes. O sinal de igual é utilizado quando dois números ou expressões são idênticos. O sinal de equivalência é usado quando dois números ou expressões possuem uma propriedade em comum. Neste caso, a propriedade em comum entre eles é o fato de os dois números inteiros possuírem o mesmo resto quando divididos por  $n$ .

### 3.4 Mapeamento Anti-simétrico

Este tópico está fortemente baseado no artigo descrito em [8].

Conforme mencionamos anteriormente, estatisticamente 90% dos erros são erros individualizados e de transposição de dígitos adjacentes. Assim sendo, um esquema de dígitos verificadores eficaz deve identificar pelo menos todos os casos de ocorrência desses dois tipos de erros, para que tenha um mínimo de confiabilidade. Por isso nosso estudo relativo à matemática dos esquemas de dígitos verificadores irá analisar principalmente os esquemas que identificam estes tipos de erros. Mas antes, precisamos definir, quais os critérios de cálculo do dígito verificador deveremos seguir para que este objetivo seja alcançado. Estes critérios são baseados na Teoria dos Grupos.

### 3.4.1 Conceitos da Teoria dos Grupos aplicados aos Dígitos Verificadores

**Definição 3.4.1.1:** Um conjunto é uma coleção de objetos chamados membros ou elementos.

**Definição 3.4.1.2:** Um subconjunto de um conjunto é um conjunto, em que cada um de seus elementos está contido no original

O método de combinar os elementos de um conjunto é chamado operação. Dado um conjunto e uma operação envolvendo dois de seus elementos, esse conjunto é fechado se o resultado também é um elemento desse conjunto. Ex: conjunto de inteiros  $Z=\{ \dots, -2, -1, 0, 1, 2, \dots \}$  associado à operação de subtração é fechada.

O conjunto de algarismos  $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$  associado à operação de adição não é fechado, já que nem todos os elementos, quando aplicada a operação, resultam em outro elemento do grupo. Ex:  $7+6=14$ , e 14 não pertence ao conjunto.

Entretanto se esse conjunto não é fechado quando associado à adição, há uma operação que o torna fechado. Se aplicarmos em quaisquer dois algarismos do grupo a operação módulo 10, citada anteriormente, o resultado sempre será um elemento do conjunto.

Esse artifício é utilizado pela maioria dos esquemas de dígitos verificadores para restringir a quantidade de algarismos passíveis de utilização e evitar que números que não possam ser representados por mais de um algarismo sejam utilizados.

Outro exemplo de conjunto fechado é o grupo diedral ( $D_{10}$ ). Se quaisquer duas simetrias de um pentágono (translação, reflexão, rotação e reflexão-deslizada) forem combinadas, o resultado será outra simetria do pentágono.

**Definição 3.4.1.3:** Se um conjunto não vazio  $G$  tem uma operação “ $*$ ” associada a ele,  $G$  é chamado de grupo quando as seguintes propriedades são encontradas:

- Fechado: Para quaisquer dois elementos  $a$  e  $b$  em  $G$ ,  $a * b$  também está em  $G$ .
- Associatividade: Para todos os elementos  $a$ ,  $b$  e  $c$  em  $G$ ,  $(a * b) * c = a * (b * c)$ .
- Identidade:  $G$  contém um elemento  $e$ , chamado identidade de  $G$ , tal que  $e * a = a * e = a$
- Inversível: Para cada elemento  $a$  em  $G$ , existe um elemento  $a^{-1}$ , chamado de inverso de  $a$ , de forma que  $a * a^{-1} = a^{-1} * a = e$ .

Cada identificador/esquema de DV possui seu grupo de caracteres aptos a serem utilizados. O ideal é que somente números ou caracteres unitários (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B ...) sejam utilizados, a fim de manter-se unitário o dígito verificador.

A Tabela 3 apresenta as condições lógicas para a detecção dos seguintes tipos de erros:

<b>Tipo de erro</b>	<b>Descrição</b>	<b>Detecção possível se</b>
<b>Erro individualizado</b>	Quando um dos dígitos constituinte do código troca de valor $a \gg b$	$T(x)$ é m permutação
<b>Transposição de dígitos adjacentes</b>	Quando dois dígitos adjacentes trocam de lugar $ab \gg ba$	$yT(x) \neq xyT(y)$ para todo $x, y \in G$ com $x \neq y$ $\square$
<b>Transposição de salto</b>	Quando dois diferentes dígitos, separados por um terceiro, trocam de posição $abc \gg cba$	$xyT^2(z) \neq zyT^2(x)$ para todo $x, y, z \in G$ com $x \neq z$

Tabela 3. Condições pra detecção dos principais tipos de erros

Veremos abaixo a explicação matemática para a permutação condicional para a identificação de todos os erros individualizados:

Uma maneira de se calcular o dígito verificador é utilizar operações em um grupo de dígitos, utilizando permutações  $t_m, \dots, t_0$ .

**Definição 3.4.1.4:** Seja  $(G, \cdot)$  um grupo finito de ordem  $n$ , e  $c$  pertencente a  $G$ , e  $m \geq 2$ . Então um esquema de dígitos verificadores sobre o grupo  $G$  é definido por  $m+1$  permutações  $t_m, \dots, t_0$  de  $G$ . Para cada número formado pelos algarismos  $x_m x_{m-1} \dots x_1$  um dígito verificador  $x_0$  é inserido, de forma que a equação de verificação  $t_m(x_m), t_{m-1}(x_{m-1}), \dots, t_1(x_1), t_0(x_0) = c$  seja satisfeita.

Para simplificar tomaremos  $G$  como o grupo  $\{0, \dots, n-1\}$  com o elemento neutro  $c=0$ .

O algarismo  $x_0$  é determinado de acordo com o identificador. Todo esquema de dígitos verificadores baseados em grupos, com permutações, detectam os erros do tipo “erro individualizado”.

### 3.4.2 Mapeamento antissimétrico

Para identificar os erros do tipo “Transposição de dígitos adjacentes”, precisamos que a permutação do esquema do dígito verificador seja antissimétrica.

**Definição 3.4.2.1:** Uma permutação  $t$  de um grupo  $(G, \cdot)$  é chamada antissimétrica, se para todo  $x, y \in G$ .

$$t(x) \cdot y = t(y) \cdot x, \text{ somente se } x=y$$

O conjunto de todos os mapeamentos antissimétricos de um grupo  $G$  são denotado como  $\text{Ant}(G)$ .

Então, um esquema de dígito verificador baseado em um grupo que detecte todos os erros do tipo “Transposição de dígitos adjacentes” existe se e somente se o grupo possuir mapeamento antissimétrico. Se  $n$  (ordem do grupo) é ímpar, então  $Z_n$  possui

mapeamento antissimétrico, já se  $n$  for par, então o grupo diedral  $D_{n/2}$  de ordem  $n$  possui o mapeamento antissimétrico.

Mapeamentos antissimétricos estão diretamente ligados aos mapeamentos completos. Uma permutação  $t$  é chamada completa se  $x \cdot t(x) = y \cdot t(y)$  implica que  $x=y$ .

Em um grupo abeliano, existe um mapeamento antissimétrico se o grupo possui um mapeamento completo.

**Corolário 3.4.2.2** (i) Um grupo de ordem  $2m$ , onde  $m$  seja ímpar, não admite um mapeamento completo.

(ii)  $Z_{10}$  não admite um sistema de dígito verificador que detecte todos os “erros individualizados” e “transposição de dígitos adjacentes”.

(iii) Assim como o EAN, nenhum outro sistema utilizando o  $Z_{10}$  é capaz de detectar todos os erros de “transposição de dígitos adjacentes”. Ou seja:

(iv) Um grupo cíclico  $G$  permite um mapeamento anti-simétrico, se e somente se  $|G|$  (quantidade de elementos do grupo) é ímpar.

(v) Grupos de ordem  $m=2u$  com  $u$  ímpar, particularmente  $D_5$  e  $Z_{10}$ , não admitem um sistema de dígito verificador que detecte todos os “erros gêmeos” e “erros gêmeos alternados”.

### 3.4.3 O Esquema de Verhoeff

Este tópico está fortemente baseado no artigo descrito em [4].

Um dos esquemas de dígito verificador que utiliza o mapeamento anti-simétrico é o esquema de Verhoeff, que o mesmo criou em sua tese de doutorado, em 1969. Ele se baseia nas propriedades de  $D_{10}$  (grupo diedral de simetrias de um pentágono regular), citado anteriormente, juntamente com permutações  $p$ , onde  $p = (0), (1,4), (2,3), (5,6,7,8,9)$ . Essas propriedades são constituídas por um sistema de operações em 10 elementos, sendo essas operações representadas pela rotação ou reflexão de um pentágono regular  $D_{10}$ . Essas operações podem ser representadas através da tabela  $x$ , onde as linhas e colunas são representadas pelos operadores da operação “\*”.

*	0	1	2	3	4	5	6	7	8	9
0	0	1	2	3	4	5	6	7	8	9
1	1	2	3	4	0	6	7	8	9	5
2	2	3	4	0	1	7	8	9	5	6
3	3	4	0	1	2	8	9	5	6	7
4	4	0	1	2	3	9	5	6	7	8
5	5	9	8	7	6	0	4	3	2	1
6	6	5	9	8	7	1	0	4	3	2
7	7	6	5	9	8	2	1	0	4	3
8	8	7	6	5	9	3	2	1	0	4
9	9	8	7	6	5	4	3	2	1	0

Tabela 4. Tabela Cayley para  $D_{10}$  de Verhoeff

#### Definição 3.4.3.1:

Seja  $a_1 a_2 \dots a_{n-1} a_n$  um identificador com dígito verificador  $a_n$ . O dígito verificador foi inserido de forma que a seguinte equação fosse satisfeita:

$$p^{n-1}(a_1) * p^{n-2}(a_2) * p^{n-3}(a_3) \dots * p(a_{n-1}) * a_n = 0$$

onde  $p$  é a comutação  $p = (0)(1,4)(2,3)(5,6,7,8,9)$ , criada por S. Winters[12] e “\*” a operação estabelecida na tabela x.

Usaremos o identificador válido “4195357”. Sendo  $a_1=4$ ,  $a_2=1$ ,  $a_3=9 \dots a_n=7$  e  $n=7$ . O último dígito é o verificador.

Para que esse identificador seja válido, a seguinte equação deve ser satisfeita:

$$p^{n-1}(a_1) * p^{n-2}(a_2) * p^{n-3}(a_3) \dots * p(a_{n-1}) * a_n = 0$$

$$p^{7-1}(a_1) * p^{7-2}(a_2) * p^{7-3}(a_3) * p^{7-4}(a_4) * p^{7-5}(a_5) * p^{7-6}(a_6) * a_7 = 0$$

$$p^6(a_1) * p^5(a_2) * p^4(a_3) * p^3(a_4) * p^2(a_5) * p^1(a_6) * a_7 = 0$$

$$p^6(4) * p^5(1) * p^4(9) * p^3(5) * p^2(3) * p^1(5) * 7 = 0$$

$$4 * 4 * 8 * 8 * 3 * 6 * 7 = 0 \quad (\text{realizando a quantidade de permutações de cada fator})$$

$$(4 * 4) * (8 * 8) * (3 * 6) * 7 = 0 \quad (\text{utilizando a tabela } D_{10})$$

$$3 * 0 * 9 * 7 = 0$$

$$(3 * 0) * (9 * 7) = 0$$

$$3 * 2 = 0$$

$$0 = 0$$

Como  $0=0$  é uma afirmação verdadeira, “4195357” é um identificador válido.

# Capítulo 4 – Esquemas de Dígitos Verificadores Difundidos no Brasil

Neste capítulo descrevemos os principais esquemas de dígitos verificadores utilizados no Brasil. Além de descrevermos e exemplificarmos cada um dos métodos apresentados, também fizemos uma análise da capacidade de detecção de erros. Essas análises foram feitas com a geração de códigos corretos, seguido de milhares de testes onde introduzimos alguns dos erros que deveriam ser corrigidos pelo esquema. Então são apresentadas tabelas com os percentuais de erros não detectados.

A seguir descrevemos cada um dos esquemas estudados.

## 4.1 Identificação de produtos por código de barras

Este tópico está fortemente baseado no artigo descrito em [10] e [11].

O Universal Product Code (UPC) foi o primeiro número identificador em código de barras amplamente utilizado nos EUA. Seu padrão foi estabelecido em abril de 1973, mas a necessidade de um padrão global para a utilização do UPC levou a criação do EAN (European Article Number ou International Article Number), em dezembro de 1976. E foi no dia 26 de junho de 1974, às 8:01 da manhã, que um cliente do supermercado Marsh's em Troy, no estado de Ohio, realizou a primeira compra de um produto com código de barras, no padrão UPC. Apesar de ter sido criado com a intenção de ser utilizado nos países europeus, logo se viu a capacidade de abranger todos os países neste mesmo padrão, facilitando o comércio mundial.

O EAN, a versão japonesa “Japanese Article Numbering” (JAN) e o International Article Numbering System (IAN) são idênticas ao UPC, exceto pelo número de dígitos. Os códigos do JAN são idênticos aos códigos do EAN, mas iniciados por 45 e 49. Visando a compatibilidade com o UPC, os códigos iniciados com 00, 01, 03, 04 e 06 a 13 são reservadas aos Estados Unidos.

Entre as 4 versões do EAN, a EAN13 é a mais utilizada pelo comércio brasileiro, sendo o código composto de 13 dígitos, distribuídos da seguinte maneira: os 3 primeiros representam o país, os 4 seguintes o código da empresa cadastrada na EAN, os outros 5 representam o código identificador do item comercial dentro da empresa e o último (13º) é o dígito verificador, obtido por meio de cálculo algoritmo, descrito posteriormente.

O padrão EAN é administrado pela GS1 (conhecida anteriormente por EAN International). Atualmente existem organizações nacionais em mais de 92 países e 5 continentes.

Os sistemas do Brasil estão modelados para receber código de 13 posições, o que possibilita trabalhar também com códigos de 8 e 12 posições, utilizados em produtos com tamanho reduzido e recipientes cilíndricos.

O código utilizado no Brasil é estipulado pela EAN usa números e seu formato é 789FFFFPPPPPV, onde 789 é o código representante do Brasil; FFFF é o código de identificação do fabricante; PPPPV identificação do produto criada pelo fabricante e V=dígito verificador.

Tendo  $G = (Z_{10}, +)$ ,  $n=13$ ,  $e=0$ ,  $\square_{2i-1}(a) = a =: L_1(a)$  e  $\square_{2i}(a) = 3a =: L_1(a)$ ; este sistema não detecta “transposição de dígitos adjacentes”  $ab \rightarrow ba$  aonde  $|b-a|=5$ , ou seja o mapeamento  $L_3L_1^{-1}$  não é anti simétrico.

Posição	1	2	3	4	5	6	7	8	9	10	11	12	13
Peso	1	3	1	3	1	3	1	3	1	3	1	3	1

Tabela 5. Cálculo código de barras EAN

Exemplo:

EAN:7891027114275

Neste caso, 789 é o código do país (Brasil), 1027 é o código do fabricante (Tilibra), 11427 é o código do produto e 5 o dígito verificador.

Para verificarmos se o dígito verificador está correto, faremos a permutação:

$$1 \times 7 + 3 \times 8 + 1 \times 9 + 3 \times 1 + 1 \times 0 + 3 \times 2 + 1 \times 7 + 3 \times 1 + 1 \times 1 + 3 \times 4 + 1 \times 2 + 3 \times 7 + 1 \times 5 =$$

$$7 + 24 + 9 + 3 + 0 + 6 + 7 + 3 + 1 + 12 + 2 + 21 + 5 = 100 \equiv 0 \pmod{10}$$

Desta forma, identificamos que o dígito verificador está correto.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela . A tabela a seguir mostra que nem todos os erros são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0%
<b>Erros alternados</b>	1000000	89,99%
<b>Erros adjacentes</b>	1000000	10,00%
<b>Erros gêmeos</b>	1000000	7,16%
<b>Erros gêmeos alternados</b>	1000000	6,86%

## 4.2 Identificação do Cadastro de Pessoas Físicas – CPF

Este tópico está fortemente baseado no artigo descrito em [12].

O CPF é um documento de identificação dos cidadãos brasileiros ou estrangeiros com negócios no país. O registro deste identificador é realizado junto a Receita Federal brasileira, sendo os dois últimos dígitos os dígitos verificadores (DVs) no esquema módulo 11. O dígito de ordem 9 indica a unidade federativa onde foi realizado dada entrada ao CPF, conforme Tabela 5.

Número	Unidades Federativas
0	Rio Grande do Sul
1	Distrito Federal, Goiás, Mato Grosso, Mato Grosso do Sul e Tocantins
2	Amazonas, Pará, Roraima, Amapá, Acre e Rondônia
3	Ceará, Maranhão e Piauí
4	Paraíba, Pernambuco, Alagoas e Rio Grande do Norte.
5	Bahia e Sergipe
6	Minas Gerais
7	Rio de Janeiro e Espírito Santo
8	São Paulo
9	Paraná e Santa Catarina

Tabela 6. Unidades federativas do CPF

O esquema módulo 11 é calculado através do somatório da multiplicação de cada um dos dígitos de um identificador (composto de 9 dígitos, antes dos DVs) por 9, 8, 7, 6, 5, 4, 3, 2, 1 e 0, sendo 9 multiplicado pelo mais a direita e assim por diante. O dígito  $a_{10}$  é escolhido de forma que a condição  $(a_1, a_2, \dots, a_{10}) \cdot (1,2,3,4,5,6,7,8,9) \equiv 0 \pmod{11}$  seja satisfeita, e o dígito  $a_{11}$  de forma que a condição  $(a_1, a_2, \dots, a_{10}, a_{11}) \cdot (0,1,2,3,4,5,6,7,8,9) \equiv 0 \pmod{11}$ . Utilizaremos o exemplo do CPF 537.171.594-02

5 3 7 1 7 1 5 9 4  
x x x x x x x x x  
1 2 3 4 5 6 7 8 9

$$\begin{aligned} & 5+6+21+4+35+6+35+72+36 = 220 \\ & 220 \equiv 0 \pmod{11} \\ & DV_1=0 \end{aligned}$$

5 3 7 1 7 1 5 9 4 0  
x x x x x x x x x  
0 1 2 3 4 5 6 7 8 9

$$0+3+14+3+28+5+30+63+32+0=178$$

$$178 \equiv 2 \pmod{11}$$

$$DV_2=2$$

Caso algum dos dígitos verificadores seja 10, ele será substituído por 0.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela . Vemos que aqui, também, alguns dos erros não são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0,17%
<b>Erros alternados</b>	1000000	0,15%
<b>Erros adjacentes</b>	1000000	0,13%
<b>Erros gêmeos</b>	1000000	0,20%
<b>Erros gêmeos alternados</b>	1000000	0,01%

### 4.3 Identificação da Carteira Nacional de Habilitação – CNH

A CNH é um documento emitido pelo DETRAN, que possui um identificador denominado número de registro, composto de 11 dígitos, sendo os 2 últimos os DV os outros o identificador em ordem sequencial. O esquema é basicamente um esquema módulo 11.

$$9*a_1+8*a_2+7*a_3+6*a_4+5*a_5+4*a_6+3*a_7+2*a_8+1*a_9 = X \Rightarrow X \pmod{11} = DV_1$$

Se  $DV_1=10$ , ele será substituído por 0.

$$1*a_1+2*a_2+3*a_3+4*a_4+5*a_5+6*a_6+7*a_7+8*a_8+9*a_9 = X \Rightarrow X \pmod{11} = DV_2$$

O segundo é calculado analogamente, mas se  $DV_1=10$ , no cálculo, antes de ser substituído por 0 ,e  $DV_2>2$ ,  $DV_2$  é subtraído de 2.

Como exemplo calcularemos os dígitos verificadores do identificador de CNH 033756375.

## DV<sub>1</sub>

0 3 3 7 5 6 3 7 5  
x x x x x x x x x  
9 8 7 6 5 4 3 2 1  
-----

$$0+24+21+42+25+24+9+14+5=164$$
$$164 \equiv 10 \pmod{11} = DV_1$$

Como DV<sub>1</sub> daria 10, o substituiremos por 0. Ou seja, DV<sub>1</sub>=0.

## DV<sub>2</sub>

0 3 3 7 5 6 3 7 5  
x x x x x x x x x  
1 2 3 4 5 6 7 8 9  
-----

$$0+6+9+28+25+36+21+56+45=226$$
$$226 \equiv 6 \pmod{11} = DV_2$$

Como no primeiro cálculo DV<sub>1</sub>=10, e nosso cálculo de DV<sub>2</sub>>2, então DV<sub>2</sub>= DV<sub>2</sub>-2. Ou seja DV<sub>2</sub>=6-2=4. DV<sub>2</sub> = 4.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela a seguir. Poucos erros permanecem não detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0,17%
<b>Erros alternados</b>	1000000	0,13%
<b>Erros adjacentes</b>	1000000	0,13%
<b>Erros gêmeos</b>	1000000	0,30%
<b>Erros gêmeos alternados</b>	1000000	0,02%

## 4.4 Identificação do Passaporte da República Federativa do Brasil.

Este tópico está fortemente baseado no artigo descrito em [13].

O passaporte da República Federativa do Brasil utiliza o padrão MRP (Machine Readable Passport), padronizado pelo Organização de Aviação Civil Internacional – OACI na década de 80, estabelecendo 3 tipos de documentos de viagens, sendo o mais comum o tipo 3, com duas linhas de 44 caracteres cada.

A primeira linha começa com a letra P(indicando que é um passaporte e não um visto), seguido de um caractere indicando o tipo de passaporte, o 3º, 4º e 5º indicam o país e o restante é o sobrenome e o nome separados por “>>”, sendo os espaços demarcados por “>”.

A segunda linha possui, nas posições de 1 a 9, o número do passaporte e o décimo o dígito verificador. Os outros dígitos são relativos à nacionalidade, data de nascimento, sexo, data de expiração, números de controle do órgão emissor (no caso a República Federativa do Brasil) e seus dígitos verificadores.

Para o cálculo do dígito verificador do número do passaporte é utilizado um esquema módulo 10, multiplicado por 7 3 1 7 3 1 7 3 1. Os algarismos de 0 a 9 são representado por eles mesmo, as letras do alfabeto são representados de 10 a 35 (incluindo k, w e y), sendo a→10, b →11 ... 35→z. Caso o nono caractere seja um “>”, o algarismo “0” será considerado.

Ex: O passaporte fictício: FO970675< deveria ser transformado em uma cadeia de algarismos.

F	O	9	7	0	6	7	5	<
=	=	=	=	=	=	=	=	(equivalência)
15	24	9	7	0	6	7	5	0
x	x	x	x	x	x	x	x	
7	3	1	7	3	1	7	3	1

---

105+72+9+49+0+6+49+24+0=314

$$314 \equiv 314 \pmod{10} = 5$$

DV=5

Sendo assim 5 é o dígito verificador do número do passaporte, ficando na posição do caractere 10.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Vemos que um percentual razoável de erros não são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	15,43%
<b>Erros alternados</b>	1000000	14,61%
<b>Erros adjacentes</b>	1000000	14,55%
<b>Erros gêmeos</b>	1000000	9,14%
<b>Erros gêmeos alternados</b>	1000000	9,37%

## 4.5 Identificação de Registro Geral.

Este tópico está fortemente baseado no artigo descrito em [12].

O identificador das carteiras de identidade, denominado Registro Geral (RG), possui órgãos emissores e cálculos diferentes para cada estado. Abordaremos aqui dois esquemas, de São Paulo (SSP-SP – Secretaria da Segurança Pública de São Paulo) e Rio de Janeiro (IFP-RJ – Instituto Felix Pacheco do Rio de Janeiro), já que são os dois maiores estados do Brasil, e o esquema que atualmente é utilizado no Rio de Janeiro, para novos registros, através do DIC - Departamento de Identificação Civil do Detran.

### SSP-SP

O identificador do RG emitido pela SSP-SP é um número seqüencial de 8 dígitos, acrescido de um nono algarismo de dígito verificador. O cálculo do dígito verificador deste órgão emissor utiliza um esquema módulo 11, onde se realiza a seguinte operação  $a_1x9+a_2x8+a_3x7 \dots a_8x2 = X \pmod{11} = a_9$ , sendo o 2 começando sempre a multiplicar o algarismo mais à esquerda do identificador. Caso  $a_9$  seja 10, utiliza-se o caractere X como dígito verificador. Ex: RG número 20932810

$$\begin{array}{r}
 2 \ 0 \ 9 \ 3 \ 2 \ 8 \ 1 \ 0 \\
 \times \ x \ x \ x \ x \ x \ x \ x \\
 \hline
 9 \ 8 \ 7 \ 6 \ 5 \ 4 \ 3 \ 2 \\
 \hline
 18+0+63+18+10+32+3+0=144
 \end{array}$$

144 ≡ 1 (mod 11)  
DV=1.

Sendo assim 1 é o dígito verificador do número do RG, ficando na posição do  $a_9$ .

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Vemos que a grande maioria dos erros são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0,0%
<b>Erros alternados</b>	1000000	0,0%
<b>Erros adjacentes</b>	1000000	0,0%
<b>Erros gêmeos</b>	1000000	6,42%
<b>Erros gêmeos alternados</b>	1000000	0,0%

## IFP-RJ e DIC-RJ

O identificador do RG emitido pela IFP-RJ ou DIC-RJ é um número seqüencial de até 8 dígitos, acrescido de um nono algarismo de dígito verificador. O cálculo do dígito verificador desses órgãos emissores utilizam um esquema módulo 10, onde os algarismos do identificador são multiplicados por 2 e 1 alternadamente, começando sempre com 2, da esquerda para direita. Caso alguma das multiplicações resulte em um valor superior a 9, soma-se os algarismos deste resultado, tornando-se esse o novo resultado. Essa operação é denominada “Noves Fora”.

Tomando como exemplo o identificador 27998162

$$\begin{array}{r}
 2\ 7\ 9\ 9\ 8\ 1\ 6\ 2 \\
 \times\ \times\ \times\ \times\ \times\ \times\ \times \\
 1\ 2\ 1\ 2\ 1\ 2\ 1\ 2 \\
 \hline
 2+14+9+18+8+2+6+4 \\
 \hline
 2+5+9+9+8+2+6+4=45 \quad (\text{após noves fora}) \\
 45 \equiv 5 \pmod{10} \\
 DV=10-5=5 \\
 DV=5
 \end{array}$$

Caso fosse igual a 10, seria substituído por 0.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Muitos erros não são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0,0%
<b>Erros alternados</b>	1000000	77,80%
<b>Erros adjacentes</b>	1000000	2,85%
<b>Erros gêmeos</b>	1000000	8,84%
<b>Erros gêmeos alternados</b>	1000000	5,18%

## 4.6 Identificação do Título Eleitor

Este tópico está fortemente baseado no artigo descrito em [12].

O Título eleitoral é emitido pelo TSE – Tribunal Superior Eleitoral, e é formado por até 8 dígitos seqüenciais de  $a_1$  até  $a_8$ , seguido por dois dígitos indicadores da Unidade da Federação (UF) de emissão, finalizando com dois caracteres de DV. Na verdade são dois dígitos verificadores calculados de formas distintas, mas ambos módulo 11. O primeiro é calculado com base no número seqüencial e o segundo é calculado com a junção do código da UF com o primeiro DV.

DV1  
8 2 4 7 4 8  
x x x x x x  
4 5 6 7 8 9  
-----

$$32+10+24+49+32+72=219  
219 \equiv 10 \pmod{11}$$

DV1=0, já que 10 é substituído por 0.

DV2  
0 3 0  
x x x  
7 8 9  
-----  
0+24+0=24  
24\equiv 2 \pmod{11}  
DV2=2

Sendo assim, o número completo do título eleitoral é 82474803/02.

As unidades da federação e seu respectivos códigos, conforme Tabela 7.

SP	MG	RJ	RS	BA	PR	CE	PE	SC	GO	MA	PB	PA	ES
01	02	03	04	05	06	07	08	09	10	11	12	13	14

PI	RN	AL	MT	MS	DF	SE	AM	RO	AC	AP	RR	TO	ZZ (Fora do país)
15	16	17	18	19	20	21	22	23	24	25	26	27	28

Tabela 7. Unidade da federação do título de eleitor

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. A grande maioria dos erros é detectada.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	0,40%
<b>Erros alternados</b>	1000000	0,32%
<b>Erros adjacentes</b>	1000000	0,32%
<b>Erros gêmeos</b>	1000000	4,83%
<b>Erros gêmeos alternados</b>	1000000	0,19%

## 4.7 Identificação do PIS / PASEP

Este tópico está fortemente baseado no artigo descrito em [14].

O PIS – Programa de Integração Social é administrado pela Caixa Econômica Federal e voltado aos trabalhadores da iniciativa privada. O PASEP – Programa de Formação do Patrimônio do Servidor Público é administrado pelo Banco do Brasil e voltado aos servidores públicos. Ambos possuem identificador com a estrutura de 10 dígitos numéricos seguido de um DV. O cálculo do dígito verificador segue o esquema módulo 11, conforme exemplo. Ex: PIS 1639723765

$$\begin{array}{cccccccccc}
 1 & 6 & 3 & 9 & 7 & 2 & 3 & 7 & 6 & 5 \\
 \times & \times \\
 3 & 2 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 \\
 \hline
 3+12+27+72+49+12+15+28+18+10=246 \\
 246 \equiv 4 \pmod{11} \\
 DV=11-4=7
 \end{array}$$

Sendo assim 7 é o dígito verificador do número do PIS/PASEP, ficando na posição  $a_{11}$ . Caso este resultado fosse 10 ou 11, o DV seria 0.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Percentuais razoáveis de erros não são detectados.

	<b>Quantidade de entradas testadas</b>	<b>Percentual de erros não identificados</b>
<b>Erros individuais</b>	1000000	9,66%
<b>Erros alternados</b>	1000000	9,58%
<b>Erros adjacentes</b>	1000000	9,58%
<b>Erros gêmeos</b>	1000000	6,24%
<b>Erros gêmeos alternados</b>	1000000	5,86%

## 4.8 Identificação da Carteira de Trabalho

Tentamos contato com a ouvidoria através do site do MTE – Ministério do Trabalho e Emprego e recebemos apenas respostas automáticas do recebimento das mensagem. Após mais de 5 meses de espera obtivemos a resposta de que a CTPS não possui dígito verificador, sendo que cada nova série, tem numeração de 0000001 à 9999999.

## 4.9 Identificação do Certificado de Reservista

Tentamos contato através do site do exército brasileiro, junto a Seção de Relações Públicas. Fomos encaminhado para a ouvidoria do DGP – Departamento Geral do Pessoal do Exército Brasileiro, que nos informou que esse tipo de informação não é repassada ao público.

## 4.10 Identificação do Cartão de Crédito

Este tópico está fortemente baseado no artigo descrito em [2].

Os identificadores dos cartões de créditos possuem entre 14 e 19 dígitos, sendo os 4 primeiros referentes ao banco emissor e bandeira do cartão. O último dígito é o dígito verificador, calculado conforme exemplo da Tabela 13, para o identificador 4312593786689.

4 3 1 2 5 9 3 7 8 6 6 8 9  
x x x x x x x x x x x x x  
2 1 2 1 2 1 2 1 2 1 2 1 2

---

8+3+2+2+10+9+6+7+16+6+12+8+18

---

8+3+2+2+1+9+6+7+7+6+3+8+9=71 (após noves fora)  
71 ≡ 1 (mod10)

DV = 10-1 = 9

Como DV=9, então 9 fica na posição do último caractere.

Caso o cálculo realizado fosse 0, o DV seria também 0.

Não foram feitas estatísticas de erros para este esquema.

#### **4.11 Identificação do Bloqueto de Cobrança**

Este tópico está fortemente baseado no artigo descrito em [15].

O bloqueto de cobrança representa títulos de cobrança de duplicatas, notas promissórias, bilhetes, recibos e outros tantos que podem ser pagos através da rede bancária.

O dígito verificador geral é baseado no código de barras, que segue o padrão da Tabela 8:

<b>Posição</b>	<b>Descrição</b>
01-03	Identificação do banco
04-04	Código da moeda (Real(R\$) - 9)
05-05	Dígito verificador do código de barras
06-19	Posições 06 a 09 – fator de vencimento Posições 10 a 19 – valor nominal do título
20-44	Campo livre – regulamentado de acordo com o banco emissor

Tabela 8. Campos do código de barras do bloqueto de cobrança

O dígito verificador geral, localizado na quinta posição, é calculado multiplicando-se da direita para a esquerda, por 2,3,4,5,6,7,8,9,2,3..., desconsiderando-se a posição 5, na qual será inserido o DV. Soma-se o resultado e calcula-se o mod 11. O DV será a diferença entre 11 e o resultado do mod 11.

Ex: 81397080300004123993577905375392379551304760

```

8 1 3 9 DV 0 8 0 3 0 0 0 0 4 1 2 3 9 9 3 5 7 7 9 0 5 3 7 5 3 9 2 3 7 9 5 5 1 3 0 4 7 6 0
x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x x
4 3 2 9 // 8 7 6 5 4 3 2 9 8 7 6 5 4 3 2 9 8 7 6 5 4 3 2 9 8 7 6 5 4 3 2 9 8 7 6 5 4 3 2 9 8 7 6 5 4 3 2

```

$32+3+6+81+0+56+0+15+0+0+0+32+7+12+15+36+27+6+45+56+49+54+0+20+9+14+45+24+63+12+15+28+27+10+45+8+21+0+20+28+18+0=939$

$$939 \equiv 4 \pmod{11}$$

$$DV=11-4=7$$

Caso o resultado do módulo 11 seja igual a 0,10 ou 11, DV=1.

O identificador é uma representação numérica do código de barras, sendo disposto em ordem diferente e acrescido de 3 dígitos verificadores.

Esse identificador é composto de 47 algarismos, divididos em 5 blocos, sendo os três primeiros com dígitos verificadores associados, além do DV geral do código de barras. O identificador segue o padrão da Tabela 9.

<b>Posição</b>	<b>Descrição</b>
01-03	Identificação do banco
04-04	Código da moeda (Real(R\$) - 9)
05-09	Cinco primeiras posições do campo livre
10-10	Dígito verificador do primeiro bloco
11-20	6 <sup>a</sup> a 15 <sup>a</sup> posições do campo livre
21-21	Dígito verificador do segundo bloco
22-31	16 <sup>a</sup> a 25 <sup>a</sup> posições do campo livre
32-32	Dígito verificador do terceiro bloco
33-33	Dígito verificador geral (posição 5 do código de barras)
34-47	Posições 34 a 37 – fator de vencimento (posições 6 a 9 do código de barras) Posições 38 a 47 – valor nominal do título (posições 10 a 19 do código de barras)

Tabela 9. Campos do identificador do bloqueto de cobrança

Cada um dos 3 primeiros blocos recebe um dígito verificador. Para calculá-los, multiplicar cada algarismo por 2,1,2,1... da direita para a esquerda e aplicar o cálculo de noves fora. Somar o resultado e calcular seu mod10. O DV será a diferença entre 10 e o resultado do mod10. O código de barras citado no exemplo acima será equivalente ao identificador da Tabela10.

1º bloco	2º bloco	3º bloco	4º bloco	5º bloco
81393.5779 5	05375.39237 9	95513.04760 5	7	08030000412399

Tabela 10. Exemplo do identificador do bloqueto de cobrança

### 1º bloco

8	1	3	9	3	5	7	7	9
x	x	x	x	x	x	x	x	x
2	1	2	1	2	1	2	1	2

---

$$\begin{aligned}
 & 16+1+6+9+6+5+14+7+18 \\
 & 7+1+6+9+6+5+5+7+9=55 \text{ (com noves fora)}
 \end{aligned}$$

$$55 \equiv 5 \pmod{10}$$

$$DV = 10 - 5 = 5$$

$$DV_1 = 5$$

## 2º bloco

0	5	3	7	5	3	9	2	3	7
x	x	x	x	x	x	x	x	x	x
1	2	1	2	1	2	1	2	1	2

---

$$0+10+3+14+5+6+9+4+3+14 \\ 0+1+3+5+5+6+9+4+3+5 = 41$$

$$41 \equiv 1 \pmod{10}$$

$$DV = 10 - 1 = 9$$

$$DV_2 = 9$$

## 3º bloco

9	5	5	1	3	0	4	7	6	0
x	x	x	x	x	x	x	x	x	x
1	2	1	2	1	2	1	2	1	2

---

$$9+10+5+2+3+0+4+14+6+0 \\ 9+1+5+2+3+0+4+5+6+0 = 35$$

$$35 \equiv 5 \pmod{10}$$

$$DV = 10 - 5 = 5$$

$$DV_3 = 5$$

Geramos randomicamente 300.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Poucos erros não são detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individualizados</b>	1000000	0,92%
<b>Erros de transposição alternada</b>	1000000	6,22%
<b>Erros de transposição de dígitos adjacentes</b>	1000000	1,01%
<b>Erros gêmeos</b>	1000000	7,13%
<b>Erros gêmeos alternados</b>	1000000	1,11%

## **4.12 Identificação de código de barras de tributos/taxas estaduais e municipais e contas de concessionárias de serviços públicos.**

Este tópico está fortemente baseado no artigo descrito em [16].

Esse identificador foi criado pelo Ceneaban “Centro Nacional de Estudos da Arrecadação Bancária”, para padronização e maior agilidade na arrecadação de tributos e pagamentos de serviços públicos. É composto por 44 algarismos, divididos em 4 blocos, sendo cada um deles terminados com um dígito verificador. O identificador segue o padrão da Tabela 11.

<b>Posição</b>	<b>Descrição</b>
01-01	Produto: Essa posição é a constante “8” para indicar que se trata de uma arrecadação.
02-02	Segmento: Identificador da empresa / órgão 1-Prefeitura 2-Saneamento 3-Energia elétrica e gás 4-Telecomunicação 5-Orgãos governamentais 6-Carnes e assemelhados ou outros 7-Multas de trânsitos 9-Uso exclusivo do banco
03-03	Identificação do valor real ou referência:  6-Valor efetivo, em reais. DV módulo 10 na quarta posição e campo valor com 11 posições; 7-Valor referência, será ajustado por índice. DV módulo 10 na quarta posição e campo valor com 11 posições; 8-Valor efetivo, em reais. DV módulo 11 na quarta posição e campo valor com 11 posições; 9-Valor referência, será ajustado por índice. DV módulo 11 na quarta posição e campo valor com 11 posições;
04-04	Dígito verificador geral
05-15	Valor:  -Caso a posição 3 indique valor efetivo (6 ou 8), este campo conterá o valor em reais. -Caso a posição 3 indique valor de referencia (7 ou 9), este campo conterá uma quantidade em moeda, zeros ou valor a ser ajustado por índice.
16-19 ou 16-23	Empresa / Orgão: Identificador de quatro dígitos controlado pela Febraban, as oito primeiras posições do cadastro geral do contribuintes do Ministério da Fazenda (CNPJ).

24-44	Campo livre para utilização da Empresa / Orgão. Caso a data de vencimento seja expressa neste campo, a mesma deverá ocupar as primeiras posições deste campo, no formato AAAAMMDD.
-------	--

Tabela 11. Campos do identificador do código de barras de serviços públicos

Há 5 dígitos verificadores no código. Um adicionado a cada bloco de 11 algarismos, mais dígito verificador geral incorporado ao código de 44 caracteres, na quarta posição.

#### **Caso a posição 3 seja “6” ou “7”:**

O DV de cada bloco é calculado multiplicando cada algarismo do bloco, por 2,1,2,1,2,1,2,1,2,1,2 da direita para a esquerda. O DV será a diferença entre 10 e a soma destes produtos (utilizando noveis fora) mod 10.

O DV geral será equivalente, mas utilizando os 43 dígitos do código.

#### **Caso a posição 3 seja “8” ou “9”:**

O DV de cada bloco é calculado multiplicando cada algarismo do bloco, por 2,3,4,5,6,7,8,9,2,3,4... da direita para a esquerda. O DV será a diferença entre 11 e a soma destes produtos mod 11. Caso a soma mod 11 seja igual a 0 ou 1, o DV será “0”

O DV geral será equivalente, mas utilizando os 43 dígitos do código.

Ex:

84890000001 0 – 12000158200 4 – 90808772331 2 – 50501112122 2

#### **Cálculo do DV Geral**

##### **Blocos 3 e 4**

9	0	8	0	8	7	7	2	3	3	1	5	0	5	0	1	1	1	2	1	2	2
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
7	6	5	4	3	2	9	8	7	6	5	4	3	2	9	8	7	6	5	4	3	2

$$63+0+40+0+24+14+63+16+21+18+4+20+0+10+0+8+7+6+10+4+6+4=339$$

$$B3\_4=339.$$

##### **Blocos 1 e 2**

8	4	8	DV	0	0	0	0	0	0	1	1	2	0	0	0	1	5	8	2	0	0
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
4	3	2	DV	9	8	7	6	5	4	3	2	9	8	7	6	5	4	3	2	9	8

$$32+12+16+0+0+0+0+0+3+2+18+0+0+0+5+20+24+4+0+0=136$$

$$B1\_2=136.$$

$$\begin{aligned}
 B1\_2 + B3\_4 &= \\
 136 + 339 &= 475 \\
 475 &\equiv 2 \pmod{11} \\
 DV_G &= 11 - 2 = 9 \\
 DV_G &= 9
 \end{aligned}$$

Caso o DV fosse 11, o mesmo seria substituído por “0”.

### **Cálculo do DV de cada bloco**

#### **Bloco 1**

$$\begin{array}{cccccccccc}
 8 & 4 & 8 & 9 & 0 & 0 & 0 & 0 & 0 & 1 \\
 x & x & x & x & x & x & x & x & x & x \\
 4 & 3 & 2 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 \\
 \hline
 32 + 12 + 16 + 81 + 0 + 0 + 0 + 0 + 0 + 2 & = 143
 \end{array}$$

$$143 \equiv 0 \pmod{11}$$

$$DV_1 = 11 - 0 = 11.$$

$$DV_1 = 0$$

Quando o cálculo do DV é 11, desconsidera-se esse valor e consideramos DV=0

#### **Bloco 2**

$$\begin{array}{cccccccccc}
 1 & 2 & 0 & 0 & 0 & 1 & 5 & 8 & 2 & 0 & 0 \\
 x & x & x & x & x & x & x & x & x & x & x \\
 4 & 3 & 2 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 \\
 \hline
 4 + 6 + 0 + 0 + 0 + 7 + 30 + 40 + 8 + 0 + 0 & = 95 \\
 95 & \equiv 7 \pmod{11} \\
 DV_2 & = 11 - 7 = 4 \\
 DV_2 & = 4
 \end{array}$$

#### **Bloco 3**

$$\begin{array}{cccccccccc}
 9 & 0 & 8 & 0 & 8 & 7 & 7 & 2 & 3 & 3 & 1 \\
 x & x & x & x & x & x & x & x & x & x & x \\
 4 & 3 & 2 & 9 & 8 & 7 & 6 & 5 & 4 & 3 & 2 \\
 \hline
 36 + 0 + 16 + 0 + 64 + 49 + 42 + 10 + 12 + 9 + 2 & = 240 \\
 240 & \equiv 9 \pmod{11} \\
 DV_3 & = 11 - 9 = 2 \\
 DV_3 & = 2
 \end{array}$$

## Bloco 4

5	0	5	0	1	1	1	2	1	2	2
x	x	x	x	x	x	x	x	x	x	x
4	3	2	9	8	7	6	5	4	3	2

$$20+0+10+0+8+7+6+10+4+6+4=75$$

$$75 \equiv 9 \pmod{11}$$

$$DV_4 = 11 - 9 = 2$$

$$DV_4 = 2$$

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Poucos erros permanecem sem detecção.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individualizados</b>	1000000	0,03%
<b>Erros de transposição alternada</b>	1000000	6,26%
<b>Erros de transposição de dígitos adjacentes</b>	1000000	0,23%
<b>Erros gêmeos</b>	1000000	0,58%
<b>Erros gêmeos alternados</b>	1000000	0,70%

## 4.13 Identificação do Cadastro Nacional de Pessoa Jurídica - CNPJ

Este tópico está fortemente baseado no artigo descrito em [12].

O CNPJ é o cadastro administrado pela Receita Federal do Brasil, para registrar informações de Pessoas Jurídicas e outras instituições semelhantes. Esse identificador é composto por 14 algarismos, sendo oito primeiros denominados “raiz” que identificam a empresa, os quatro subsequentes o “sufixo” que identificam a unidade comercial produtiva dessa empresa, seguidos pelo dígito verificador, esquema módulo 11, que é composto de dois algarismos

O dígito verificador é calculado através do somatório da multiplicação de cada um dos dígitos do identificador (raiz + sufixo), composto de 12 algarismos, por 9, 8, 7, 6, 5, 4, 3 e 2, sendo 9 multiplicado pelo mais a direita e assim por diante. O dígito  $a_{13}$  é escolhido de forma que  $a_{13}=DV_1= ((a_1, a_2, \dots, a_{12}) . (6, 7, 8, 9, 2, 3, 4, 5, 6, 7, 8, 9)) \text{ mod } 11$  e o dígito  $a_{14}$  escolhido de forma que  $a_{14}=DV_2= ((a_1, a_2, \dots, a_{13}) . (5, 6, 7, 8, 9, 2, 3, 4, 5, 6, 7, 8, 9)) \text{ mod } 11$

mod11. Em ambos os casos, caso o resultado seja maior ou igual a 10, o dígito verificador será 0. Utilizaremos o exemplo do CNPJ 03.749.639/0001.

### Dígito Verificador 1

$$\begin{array}{cccccccccc}
 0 & 3 & 7 & 4 & 9 & 6 & 3 & 9 & 0 & 0 & 0 & 1 \\
 x & x & x & x & x & x & x & x & x & x & x & x \\
 6 & 7 & 8 & 9 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
 \hline
 0+21+56+36+18+18+12+45+0+0+0+9= \\
 215 \equiv 6 \pmod{11} \\
 DV_1=6
 \end{array}$$

### Dígito Verificador 2

$$\begin{array}{cccccccccc}
 0 & 3 & 7 & 4 & 9 & 6 & 3 & 9 & 0 & 0 & 0 & 1 & 6 \\
 x & x & x & x & x & x & x & x & x & x & x & x & x \\
 5 & 6 & 7 & 8 & 9 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
 \hline
 0+18+49+32+81+12+9+36+0+0+0+8+54=299 \\
 299 \equiv 2 \pmod{11} \\
 DV_2=2
 \end{array}$$

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Poucos erros permanecem não detectados.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individualizados</b>	1000000	0,03%
<b>Erros de transposição alternada</b>	1000000	0,12%
<b>Erros de transposição de dígitos adjacentes</b>	1000000	0,11%
<b>Erros gêmeos</b>	1000000	0,45%
<b>Erros gêmeos alternados</b>	1000000	0,01%

## 4.14 Identificação de agência e conta bancária – Banco do Brasil

Este tópico está fortemente baseado no artigo descrito em [17].

As instituições bancárias normalmente utilizam um esquema módulo 11, sendo o mesmo para o DV da agência e da conta. Para o Banco do Brasil identificamos um esquema no qual os algarismos do identificador são multiplicados por 2,3,4... n, da direita para esquerda, e em sua soma é aplicado o módulo 11. O DV é a diferença entre o valor encontrado e 11. Utilizaremos o exemplo da agência 0025 e conta 1023183.

Agência:

0 0 2 5

x x x x

5 4 3 2

-----

$$0+0+6+10 = 16$$

$$16 \pmod{11} \equiv 5 \Rightarrow DV=11-5=6$$

Conta:

1 0 2 3 1 8 3

x x x x x x x

8 7 6 5 4 3 2

-----

$$8+0+12+15+4+16+3=51$$

$$51 \pmod{11} \equiv 7 \Rightarrow DV=11-7=4$$

Caso o resultado do cálculo do dígito verificador seja 10, o mesmo será representado pelo caractere X.

Geramos randomicamente 1.000.000 de identificadores únicos e simulamos os cinco principais erros para cada um deles, a fim de obter uma estatística da eficiência do esquema na identificação de erros, conforme tabela. Poucos erros permanecem sem detecção.

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	1000000	1,63%
<b>Erros alternados</b>	1000000	1,46%
<b>Erros adjacentes</b>	1000000	1,49%
<b>Erros gêmeos</b>	1000000	7,87%
<b>Erros gêmeos alternados</b>	1000000	0,44%

## 4.15 Identificação de agência e conta bancária – Bradesco

Este tópico está fortemente baseado no artigo descrito em [17].

As instituições bancárias normalmente utilizam um esquema módulo 11, sendo o mesmo para o DV da agência e da conta. Para o Bradesco identificamos um esquema no qual os algarismos do identificador são multiplicados por 2,3,4... n, da direita para esquerda, e em sua soma é aplicado o módulo 11. O DV é a diferença entre o valor encontrado e 11. Utilizaremos o exemplo da agência 0412 e conta 27193.

Agência:

0 4 1 2  
x x x x  
5 4 3 2  
-----

$$0+16+3+4=23$$

$$23 \pmod{11} \equiv 1 \Rightarrow DV = 11 - 1 = 10$$

Nesse caso, como o DV possui dois algarismos, o mesmo é substituído por 0. Caso fosse uma conta do tipo poupança, seria substituído por P.

Conta:

2 7 1 9 3  
x x x x x  
6 5 4 3 2  
-----

$$12+35+4+27+6=74$$

$$74 \pmod{11} \equiv 8 \Rightarrow DV = 11 - 8 = 3$$

Geramos todas as combinações de erros estudados e testamos se o dígito verificador identificava o erro gerado, a fim de obter uma estatística de falha. Obtivemos o resultado interessante de que todos os erros foram detectados.:

	Quantidade de entradas testadas	Percentual de erros não identificados
<b>Erros individuais</b>	70	0%
<b>Erros alternados</b>	7	0%
<b>Erros adjacentes</b>	6	0%
<b>Erros gêmeos</b>	10	0%
<b>Erros gêmeos alternados</b>	10	0%

## 4.16 Identificação de agência e conta bancária – Caixa Econômica Federal

Este tópico está fortemente baseado no artigo descrito em [18].

Diferentemente do Banco do Brasil e do Bradesco, a Caixa Econômica Federal utiliza um único dígito verificador englobando o PV (uma espécie de agência), operação (valores diferentes pra conta corrente, poupança, investimento e etc) e identificador da conta. Eles são separados por “.” Sendo quatro dígitos do PV, três da operação e oito da conta. É utilizado um esquema módulo 11 no qual os algarismos do identificador são multiplicados por 2,3,4... n, da direita para esquerda, e em sua soma é aplicado o módulo 11. O DV é a diferença entre o valor encontrado e 11. Utilizaremos no exemplo o identificador 0127.001.00041482, onde 0127 é o PV, 001 a operação e 00041482 a conta.

0	1	2	7	0	0	1	0	0	0	4	1	7	8	2
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
8	7	6	5	4	3	2	9	8	7	6	5	4	3	2

---

$$\begin{aligned} & 0+7+12+35+0+0+2+0+0+24+5+28+24+4=141 \\ & 141 \text{ (mod 11)} \equiv 9 \Rightarrow DV=11-9=2 \end{aligned}$$

Caso o resultado do cálculo fosse 10, o mesmo seria substituído por 0, a fim de evitar um verificador de dois dígitos.

# Capítulo 5 – Implementação de site

A fim de complementar o Projeto Final de Dígitos Verificadores, criamos um site com informações didáticas dos dígitos verificadores citados nesse trabalho, sendo possível inclusive checar se identificadores e seu respectivos estão corretos, junto com o cálculo realizado para cada caso. O site foi desenvolvido em linguagem JAVA e a biblioteca das funções utilizadas para os cálculos está disponível no site. Este capítulo mostra um esboço desse site.

## 5.1 Pagina Inicial

O site tem uma página inicial de identificação, que dá algumas informações gerais e relaciona, em um menu na esquerda, os esquemas de dígitos verificadores a serem mostrados. A tela inicial é a seguinte:

Página Inicial

**Identificadores**

- Banco do Brasil
- Bradesco
- CEF - Caixa Econômica Federal
- Boleto de Tributos
- Boleto de Cobrança
- Cartão de Crédito
- CNH - Carteira Nacional de Habilitação
- CNPJ - Cadastro Nacional de Pessoa Jurídica
- CPF - Cadastro Nacional de Pessoa Física
- EAN - código de barras
- Passaporte**
- PIS / PASEP
- RG - RJ
- RG - SP

**Dígitos verificadores**

Dígitos verificadores são mecanismos que utilizam um ou mais dígitos (numéricos ou alfanuméricos), acrescentados a uma cadeia de dígitos (numérica ou alfanumérica) original, que certifica e/ou corrige esta cadeia, dando maior segurança contra fraudes, erros de digitação ou leitura (através de um scanner, por exemplo). Esses dígitos são formulados através de algoritmos que podem ser públicos ou não. Tais dígitos são utilizados em códigos de barra de produtos, números de conta corrente, documentos de identidade, CPF, CNPJ, etc. com as finalidades mencionadas anteriormente e têm grande importância no mundo comercial, na identificação civil, arrecadação de tributos, instituições financeiras e muitas outras áreas, onde seja vantoso. Esse site tem como objetivo por em prática, de forma didática, os conceitos apresentados no Projeto Final de Graduação dos alunos Daniel Argenta e Rafael Amorim.

## 5.2 Páginas de demonstração

Para cada opção mostrada no menu da esquerda, o site abre uma nova página, onde é feita uma exemplificação do esquema utilizado. É também disponibilizado um campo para teste por parte do usuário.

Quando o usuário tecla um determinado identificador, o sistema dá uma mensagem de estar correto ou não o identificador e justifica, passo a passo, a mensagem dada. A seguir apresentamos cada uma das telas relativas a esses esquemas. A descrição segue aquela do site, que é diferente da ordem apresentada no Capítulo 4.

## 5.2.1 Conta Bancária - Banco do Brasil

The screenshot shows a web browser window titled "Banco do Brasil". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below this, the section title "Conta Bancária - Banco do Brasil" is displayed. A text instruction reads: "Insira abaixo número de agência ou conta, ou utilize o exemplo, e confirme o cálculo realizado." Below the instruction is a text input field and a "Validar" button. To the right of the input field, there is explanatory text about the mod 11 validation scheme for Banco do Brasil accounts, mentioning the example 0025 and 1023183. Below this, there is a detailed calculation for the account 1023183-4, showing the steps: 0+0+6+10=16, 16 (mod 11) = 5 => DV=11-5=6. The account number is shown as 10 2 3 1 8 3, with intermediate steps: xx x x x x x, 87 6 5 4 3 2. The final result is 8+0+12+15+4+16+3=61, 61 (mod 11) = 7 => DV=11-7=4. A note states that if the result is 10, it is represented by 'X'. The browser status bar at the bottom shows "Done", "Internet", "100%", and other standard icons.

## 5.2.2 Conta Bancária - Bradesco

The screenshot shows a web browser window titled "Conta Bancária - Bradesco". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below this, the section title "Conta Bancária - Bradesco" is displayed. A text instruction reads: "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." Below the instruction is a text input field and a "Validar" button. To the right of the input field, there is explanatory text about the mod 11 validation scheme for Bradesco accounts, mentioning the example 0412 and 27193. Below this, there is a detailed calculation for the account 0412, showing the steps: 0+4+1+2=7, 7 (mod 11) = 7 => DV=11-7=4. The account number is shown as 0 4 1 2, with intermediate steps: x x x x, 5 4 3 2. The final result is 0+16+3+4=23, 23 (mod 11) = 1 => DV=11-1=10. A note states that if the DV has two digits, the second digit is replaced by 0. Below this, there is a calculation for the account 27193, showing the steps: 2+7+1+9+3=26, 26 (mod 11) = 3 => DV=11-3=8. The account number is shown as 2 7 1 9 3, with intermediate steps: x x x x x. The browser status bar at the bottom shows "Done", "Internet", "100%", and other standard icons.

## 5.2.3 Conta Bancária - Caixa Econômica Federal

The screenshot shows a web page titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". The main content is about the Caixa Econômica Federal's bank account digit verification system. It explains that the identifier (containing PV, operation, and account number) is multiplied by weights (2, 3, 4, 5, 6, 7, 8, 9) from right to left, and the sum is taken modulo 11. If the result is 10, it is replaced by 0. An example is given: 012700100041782, where the identifier is 01270010004178 and the operation is 2. The calculation is:  $0 \cdot 2 + 1 \cdot 3 + 2 \cdot 4 + 7 \cdot 5 + 0 \cdot 6 + 0 \cdot 7 + 0 \cdot 8 + 1 \cdot 9 = 41$ . Since  $41 \equiv 9 \pmod{11}$ , the DV is 2. A note says that if the result was 10, it would be replaced by 0 to avoid a two-digit verifier.

## 5.2.4 Boletos de Tributos e Serviços Públicos (Telefone, Luz, Água ...)

The screenshot shows a web page titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". The main content is about the Ceneaban's public service bill digit verification system. It explains that the identifier is composed of 44 digits divided into 4 blocks, with a check digit at the end. A table provides the mapping for the first two digits of the identifier:

Posição	Descrição
01-01	Produto: Essa posição é a constante "8" para indicar que se trata de uma arrecadação. Segmento: Identificador da empresa / órgão 1-Prefeitura 2-Saneamento 3-Energia elétrica e gás 4-Telecomunicação 5-Orgãos governamentais 6-Carnes e assemelhados ou outros 7-Multas de trânsitos 9-Uso exclusivo do banco
02-02	Identificação do valor real ou referência

## 5.2.5 Boleto de cobrança (página 1)

Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim

← Voltar

### Boleto de cobrança

Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado.

Posição	Descrição
01-03	Identificação do banco
04-04	Código da moeda (Real(R\$) - 9)
05-05	Dígito verificador do código de barras
06-19	Posições 06 a 09 – fator de vencimento
Posições 10 a 19	– valor nominal do título
20-44	Campo livre – regulamentado de acordo com o banco emissor

O bloqueto de cobrança representa títulos de cobrança de duplicatas, notas promissórias, bilhetes, recibos e outros tantos que podem ser pagos através da rede bancária.

O dígito verificador geral é baseado no código de barras, que segue o padrão abaixo:

8 1 3 0 DV 0 8 0 3 0 0 0 0 1 1 2 3 0 0 3 5 7 7 9 0 5 3 7 5 3 0 2 3 7 0 5 5 1 3 0 1 7 6 0

## 5.2.5 Boleto de cobrança (página 2)

Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim

← Voltar

### Boleto de cobrança

Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado.

Posição	Descrição
01-03	Identificação do banco
04-04	Código da moeda (Real(R\$) - 9)
05-05	Dígito verificador do código de barras
06-19	Posições 06 a 09 – fator de vencimento
Posições 10 a 19	– valor nominal do título
20-44	Campo livre – regulamentado de acordo com o banco emissor

O bloqueto de cobrança representa títulos de cobrança de duplicatas, notas promissórias, bilhetes, recibos e outros tantos que podem ser pagos através da rede bancária.

O dígito verificador geral é baseado no código de barras, que segue o padrão abaixo:

8 1 3 0 DV 0 8 0 3 0 0 0 0 1 1 2 3 0 0 3 5 7 7 9 0 5 3 7 5 3 0 2 3 7 0 5 5 1 3 0 1 7 6 0

## 5.2.6 Cartão de Crédito

The screenshot shows a web browser window titled "Boleto de cobrança". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below this, there is a link "← Voltar". The main section is titled "Boleto de cobrança". It contains the following text: "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." followed by a text input field and a "Validar" button. Below this, it says: "O bloqueto de cobrança representa títulos de cobrança de duplicatas, notas promissórias, bilhetes, recibos e outros tantos que podem ser pagos através da rede bancária." and "O dígito verificador geral é baseado no código de barras, que segue o padrão abaixo:". A table follows:

Posição	Descrição
01-03	Identificação do banco
04-04	Código da moeda (Real(R\$) - 9)
05-05	Dígito verificador do código de barras
06-19	Posições 06 a 09 – fator de vencimento
20-24	Posições 10 a 19 – valor nominal do título
20-24	Campo livre – regulamentado de acordo com o banco emissor

Below the table, it says: "O dígito verificador geral, localizado na quinta posição, é calculado multiplicando-se da direita para a esquerda, por 2,3,4,5,6,7,8,9,2,3..., desconsiderando-se a posição 5, na qual será inserido o DV. Soma-se o resultado e calcula-se o mod 11. O DV será a diferença entre 11 e o resultado do mod 11. Ex: 8 1 3 0 DV 0 8 0 3 0 0 0 0 1 1 2 3 0 0 3 5 7 7 0 0 5 3 7 5 3 0 2 3 7 0 5 5 1 3 0 1 7 6 0". The browser status bar at the bottom right shows "Internet" and "100%".

## 5.2.7 CNH – Carteira Nacional de Habilitação

The screenshot shows a web browser window titled "CNH - Carteira Nacional de Habilitação". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below this, there is a link "← Voltar". The main section is titled "CNH - Carteira Nacional de Habilitação". It contains the following text: "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." followed by a text input field and a "Validar" button. Below this, it says: "A CNH é um documento emitido pelo DETRAN, que possui um identificador denominado número de registro, composto de 11 dígitos, sendo os 2 últimos os DV os outros o identificador em ordem seqüencial. O esquema é basicamente um esquema módulo 11." and "9\*a1+8\*a2+7\*a3+6\*a4+5\*a5+4\*a6+3\*a7+2\*a8+1\*a9 = X => X mod11= DV1". It also states: "Se DV1=10, ele será substituído por 0." and "1\*a1+2\*a2+3\*a3+4\*a4+5\*a5+6\*a6+7\*a7+8\*a8+9\*a9 = X => X mod11 = DV2". Below this, it says: "O segundo é calculado analogamente, mas se DV1=10, no cálculo, antes de ser substituído por 0, e DV2>2, DV2 é subtraído de 2. Como exemplo calcularemos os dígitos verificadores do identificador de CNH 043243843." A table follows:

DV1	0 3 3 7 5 6 3 7 5
	XXXXXX
	9 8 7 6 5 4 3 2 1

The browser status bar at the bottom right shows "Internet" and "100%".

## 5.2.8 CNPJ – Cadastro Nacional de Pessoas Jurídicas

The screenshot shows a web browser window for the CNPJ - Cadastro Nacional de Pessoas Jurídicas. The title bar says "CNPJ - Cadastro Nacional de Pessoas Jurídicas". The main content area has a header "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below it is a link "← Voltar". The main section is titled "CNPJ - Cadastro Nacional de Pessoas Jurídicas" and contains the instruction "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." followed by a "Validar" button. A detailed explanation of the DV13 calculation is provided:

O CNPJ é o cadastro administrado pela Receita Federal do Brasil, para registrar informações de Pessoas Jurídicas e outras instituições semelhantes. Esse identificador é composto por 14 algarismos, sendo oito primeiros denominados "raiz" que identificam a empresa, os quatro subsequentes o "sufixo" que identificam a unidade comercial produtiva dessa empresa, seguidos pelo dígito verificador, esquema módulo 11, que é composto de dois algarismos.

O dígito verificador é calculado através do somatório da multiplicação de cada um dos dígitos do identificador (raiz + sufixo), composto de 12 algarismos, por 9, 8, 7, 6, 5, 4, 3 e 2, sendo 9 multiplicado pelo mais à direita e assim por diante. O dígito a13 é escolhido de forma que  $a13 = DV1 = ((a1, a2, \dots, a12) \cdot (6, 7, 8, 9, 2, 3, 4, 5, 6, 7, 8, 9)) \text{ mod } 11$ . mod11 e o dígito a14 escolhido de forma que  $a14 = DV2 = ((a1, a2, \dots, a13) \cdot (5, 6, 7, 8, 9, 2, 3, 4, 5, 6, 7, 8, 9)) \text{ mod } 11$ . Em ambos os casos, caso o resultado seja maior ou igual a 10, o dígito verificador será 0. Utilizaremos o exemplo do CNPJ 03.749.639/0001.

Dígito Verificador 1  
0 3 7 4 9 6 3 9 0 0 0 1  
x x x x x x x x x x x x  
6 7 8 9 2 3 4 5 6 7 8 9  
-----  
0+21+56+36+18+18+12+45+0+0+9= 215 = 6 (mod 11)  
DV1=6

## 5.2.9 CPF – Cadastro de Pessoas Físicas

The screenshot shows a web browser window for the CPF - Cadastro Nacional de Pessoas Físicas. The title bar says "CPF - Cadastro Nacional de Pessoas Físicas". The main content area has a header "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". Below it is a link "← Voltar". The main section is titled "CPF - Cadastro Nacional de Pessoas Físicas" and contains the instruction "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." followed by a "Validar" button. A detailed explanation of the CPF structure is provided:

O CPF é um documento de identificação dos cidadãos brasileiros ou estrangeiros com negócios no país. O registro deste identificador é realizado junto a Receita Federal brasileira, sendo os dois últimos dígitos os dígitos verificadores (DVs) no esquema módulo 11 e o anterior a esses relativo a a unidade federativa onde foi realizado a entrada ao CPF, conforme abaixo:

Número	Unidades Federativas
0	Rio Grande do Sul
1	Distrito Federal, Goiás, Mato Grosso, Mato Grosso do Sul e Tocantins
2	Amazonas, Pará, Roraima, Amapá, Acre e Rondônia
3	Ceará, Maranhão e Piauí
4	Paraíba, Pernambuco, Alagoas e Rio Grande do Norte.
5	Bahia e Sergipe
6	Minas Gerais
7	Rio de Janeiro e Espírito Santo
8	São Paulo
9	Pernambuco, Ceará, Piauí, Bahia, Minas Gerais, Rio de Janeiro, São Paulo, Paraná, Santa Catarina e Rio Grande do Sul

## 5.2.10 Código de Barras - EAN (European / International Article Number)

Código de barras - EAN (European Article Number)

Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim

← Voltar

### Código de barras - EAN (European Article Number)

Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado.

O Universal Product Code (UPC) foi o primeiro número identificador em código de barras amplamente utilizado nos EUA. Seu padrão foi estabelecido em abril de 1973, mas a necessidade de um padrão global para a utilização do UPC levou a criação do EAN, em dezembro de 1976. Foi no dia 26 de junho de 1974, às 8:01 da manhã, que um cliente do supermercado Marsh's em Troy, no estado de Ohio, realizou a primeira compra de um produto com código de barras, no padrão UPC. Apesar de ter sido criado com a intenção de ser utilizados nos países europeus, logo se viu a capacidade de abranger todos os países neste mesmo padrão, facilitando o comércio mundial.

O EAN, a versão japonesa "Japanese Article Numbering" (JAN) e o International Article Numbering System (IAN) são idênticas ao UPC, exceto pelo número de dígitos. Os códigos do JAN são idênticos aos códigos do EAN, mas iniciados por 45 e 49. Visando a compatibilidade com o UPC, os códigos iniciados com 00, 01, 03, 04 e 06 a 13 são reservadas aos Estados Unidos.

Entre as 4 versões do EAN a EAN13 é a mais utilizada pelo comércio brasileiro, sendo o código composto de 13 dígitos, distribuídos da seguinte maneira: os 3 primeiros representam o país, os 4 seguintes o código da empresa cadastrada na EAN, os outros 5 representam o código identificador do item comercial dentro da empresa e o último (13º) é o dígito verificador, obtido por meio de cálculo algorítmico, descrito posteriormente.

O padrão EAN é administrado pela GS1 (conhecida anteriormente por EAN International). Atualmente existem organizações nacionais em mais de 92 países e 5 continentes.

Os sistemas do Brasil estão modelados para receber código de 13 posições, o que possibilita trabalhar também com códigos de 8 e 12 posições, utilizados em produtos com tamanho reduzido e recipientes cilíndricos.

## 5.2.11 Passaporte

Passaporte

Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim

← Voltar

### Passaporte

Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado.

O passaporte da República Federativa do Brasil utiliza o padrão MRP (Machine Readable Passport), padronizado pela Organização de Aviação Civil Internacional – OACI na década de 80, estabelecendo 3 tipos de documentos de viagens, sendo o mais comum o tipo 3, com duas linhas de 44 caracteres cada.

A primeira linha começa com a letra P (indicando que é um passaporte e não um visto), seguido de um caractere indicando o tipo de passaporte, o 3º, 4º e 5º indicam o país e o restante é o sobrenome e o nome separados por '>>', sendo os espaços demarcados por '>'.

A segunda linha possui, nas posições de 1 a 9, o número do passaporte e o décimo o dígito verificador. Os outros dígitos são relativos à nacionalidade, data de nascimento, sexo, data de expiração, números de controle do órgão emissor (no caso a República Federativa do Brasil) e seus dígitos verificadores.

Para o cálculo do dígito verificador do número do passaporte é utilizado um esquema módulo 10, multiplicado por 7 3 1 7 3 1 7 3 1. Os algarismos de 0 a 9 são representado por eles mesmos, as letras do alfabeto são representados de 10 a 35 (incluindo k, w e y), sendo a→10, b→11 ... 35→z. Caso o nono caractere seja um ">", o algarismo "0" será considerado.

Ex: O passaporte fikticio: FO970675< deveria ser transformado em uma cadeia de algarismos.

F O 9 7 0 6 7 5 <  
= = = = = = = = (equivalência)

## 5.2.12 PIS/PASEP

The screenshot shows a web browser window with the title "PIS / PASEP". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". It contains a form field with placeholder text "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." and a "Validar" button. Below the form, there is explanatory text about the PIS and PASEP programs, their administrators, and the 10-digit identifier structure. It includes a sample number (1639723765) followed by several asterisks, and a calculation example:  $3+12+27+72+49+12+15+28+18+10=246$ ,  $246 \equiv 4 \pmod{11}$ , DV=11-4=7. A note states that 7 is the check digit for the PIS/PASEP number. The browser status bar at the bottom right shows "Internet" and "100%".

## 5.2.13 RG-RJ

The screenshot shows a web browser window with the title "RG - Registro Geral de identidade - Rio de Janeiro". The main content area is titled "Projeto Final: Dígitos Verificadores - Daniel Argenta & Rafael Amorim". It contains a form field with placeholder text "Insira abaixo os valores citados ou outro de sua preferência e confirme o cálculo realizado." and a "Validar" button. Below the form, there is explanatory text about the RG (Registro Geral) and its issuers in Rio de Janeiro (IFP-RJ and DETRAN-RJ). It describes the 8-digit identifier structure and the "Noves Fora" calculation method. A sample number (27998162) is shown with several asterisks, and a calculation example:  $2+14+9+18+8+2+6+4 = 45$ ,  $45 \equiv 5 \pmod{10}$ . The browser status bar at the bottom right shows "Internet" and "100%".

## 5.2.14 RG-SP

The screenshot shows a web page titled "RG - Registro Geral de identidade - São Paulo". The main content is about calculating the digit verifier for an RG number. It includes a text input field, a "Validar" button, and a detailed explanation of the calculation process for the number 20932810.

O identificador das carteiras de identidade, denominado Registro Geral (RG), possui órgãos emissores e cálculos diferentes para cada estado. Abordaremos aqui o esquema de São Paulo (SSP-SP – Secretaria da Segurança Pública de São Paulo), que é um dos maiores estados do Brasil.

O identificador do RG emitido pela SSP-SP é um número seqüencial de 8 dígitos, acrescido de um nono algarismo de dígito verificador. O cálculo do dígito verificador desse órgão emissor utiliza um esquema módulo 11, onde se realiza a seguinte operação  $a_1x_9 + a_2x_8 + a_3x_7 \dots a_8x_2 = X \pmod{11}$ , sendo o 2 começando sempre a multiplicar o algarismo mais à esquerda do identificador. Caso  $a_9$  seja 10, utiliza-se o caractere X como dígito verificador. Ex: RG número 20932810

2 0 9 3 2 8 1 0  
XXXXXX  
9 8 7 6 5 4 3 2  
-----  
18+0+63+18+10+32+3+0=144  
144 ≡ 1 (mod 11)  
DV=1.

Sendo assim 1 é o dígito verificador do número do RG, ficando na posição do a9.

## 5.2.15 Título de Eleitor

The screenshot shows a web page titled "Título de Eleitor". The main content is about calculating the digit verifiers for a voter title. It includes a text input field, a "Validar" button, and a detailed explanation of the calculation process for the number 824748.

O Título eleitoral é emitido pelo TSE – Tribunal Superior Eleitoral, e é formado por até 8 dígitos sequenciais de a1 até a8, seguido por dois dígitos indicadores da Unidade da Federação (UF) de emissão, finalizando com dois caracteres de DV. Na verdade são dois dígitos verificadores calculados de formas distintas, mas ambos módulo 11. O primeiro é calculado com base no número sequencial e o segundo é calculado com a junção do código da UF com o primeiro DV.

DV1  
8 2 4 7 4 8  
XXXXXX  
4 5 6 7 8 9  
-----  
32+10+24+49+32+72=219  
219 ≡ 10 (mod 11)

DV1=0, já que 10 é substituído por 0.

DV2  
0 3 0  
XXX

# Capítulo 6 - Conclusão

Neste trabalho apresentamos a conceituação de dígitos verificadores e as bases matemáticas para sua utilização.

Apresentamos também quatorze dos principais esquemas utilizados no Brasil para esse fim. Para cada um desses esquemas fizemos sua descrição, apresentamos informações complementares e fizemos, para quase todos eles, um estudo estatístico do real poder de detecção encontrado. Esse estudo prático mostrou que nem sempre os esquemas são utilizados de forma perfeita, pois alguns erros não são detectados.

Observamos que, apesar de ser matematicamente possível a utilização de esquemas de dígitos verificadores que identifiquem diversos tipos de erros, os mais utilizados continuam sendo os que detectam 100% dos erros individualizados e pequena parte de transposição de dígitos adjacentes. Provavelmente os novos identificadores que estão sendo criados para o novo padrão de identidade RIC (Registro Único de Identidade Civil) e as novas cédulas da moeda real (R\$) possuam dígitos verificadores mais eficientes. Quem sabe até dígitos corretores de erros.

Nos estudos realizados para a realização desse trabalho foi muito difícil conseguir os esquemas dos dígitos verificadores, e em alguns casos nem mesmo o órgão responsável pode nos informar o esquema, por desconhecimento ou confidencialidade.

Desenvolvemos também um site que, além de descrever didaticamente como são formados os diversos dígitos verificadores, também permite ao usuário testes para a validação de identificadores. Esses testes servem tanto para uma melhor compreensão dos esquemas utilizados quanto para a utilização prática de verificações dos identificadores.

Desta forma julgamos ter contribuído para um melhor entendimento desses importantes elementos utilizados na vida dos cidadãos brasileiros.

# Referências

- [1] Damm, Michael. "Check digit systems over groups and anti-symmetric mapping", 2000.
- [2] Gallian, J.A. "Error Detection Methods", University of Minnesota, 1996.
- [3] Verhoeff, J. "*Error Detecting Decimal Codes*", Mathematical Centre Tract 29, The Mathematical Centre, Amsterdam, 1969.
- [4] Kirtland, Joseph. "Identification Numbers and Check Digit Schemes", The Mathematical Association of America".
- [5] Oliveira Santos, Ana Carla. "Tolerância a Falhas para Sistemas Embarcados", UFPE, Recife, 2000.
- [6] Vera Pless. "*Introduction to the Theory of error-correcting codes*". New York: Wiley, 1982.
- [7] Hill, Raymond. "A First Course in Coding Theory", Oxford University Press, Oxford, 1986.
- [8] Ralph-Hardo Schulz, "*Check Character Systems and Anti-symmetric Mappings*". Departament of Mathematics and Computer Science, Free University of Berlin, 2001.
- [9] Winters, S.J., Error Detecting Schemes Using Dihedral Groups, UMAP Journal, 11, 1990, 299-308.
- [10] Economy-point.org. "<http://www.economy-point.org/e/european-article-number.html>" em 09/03/2010.
- [11] "Adams 1" website. <http://adams1.com/upcode.html>
- [12] "Telmo Ghiorzi" website. <http://www.ghiorzi.org/cgcancpf.htm>
- [13] "Alan De Smet" website. <http://www.highprogrammer.com/alan/numbers/mrp.html>
- [14] Andressa Regina Frangullys, Soraia Luciana Iazynski, Stella Maris Klemz Costa, "PIS/PASEP", FESP - FUNDAÇÃO DE ESTUDOS SOCIAIS DO PARANÁ, 2009.
- [15] Caixa, Especificação do Código de Barras Para Bloquetos de Cobrança Sem Registro SICOB. Disponível em [http://downloads.caixa.gov.br/\\_arquivos/cobrcaixasicob/manuaissicob/ESPCODBARBLOQ\\_COBRANREGIST\\_16POSICOES.pdf](http://downloads.caixa.gov.br/_arquivos/cobrcaixasicob/manuaissicob/ESPCODBARBLOQ_COBRANREGIST_16POSICOES.pdf)

[16] Febraban, “Layout” Padrão de Arrecadação/Recebimento com Utilização do Código de Barras - VERSÃO 04. Disponível em <http://www.febraban.org.br>

[17] [http://pt.wikipedia.org/wiki/D%C3%ADgito\\_verificador](http://pt.wikipedia.org/wiki/D%C3%ADgito_verificador)

[18] ATO DECLARATÓRIO EXECUTIVO Nº 1, DE 23 DE MAIO DE 2002 – Especificação de Leiaute de Registro Detalhe Referente ao Documento para Depósitos Judiciais e Extrajudiciais à Ordem e à Disposição da Autoridade Judicial ou Administrativa Competente (DJE).