

0.1 Exercício de Algoritmos com Seleção: Lógica e Algoritmo

Trabalho de Implementação em Python, em **dupla**.

Data de entrega: 17/11/2015

0.1.1 Introdução

Segundo¹ estatísticas, 90% dos erros cometidos por aqueles que precisam digitar grandes quantidades de números extensos (por exemplo, vários números de RG) são de dois tipos: erros singulares (digitase apenas um algarismo errado, como 7328 em vez de 7326) ou de transposição (troca-se a ordem de um par de algarismos, por exemplo, registra-se 9465 em vez de 9456). Para identificarem erros de um desses dois tipos, os sistemas modernos de informação propõem o acréscimo de um dígito de controle capaz de identificar se o número digitado contém ou não algum erro. A ideia de implementar sistemas de identificação com dígitos detectores de erros encontra diversas aplicações no nosso cotidiano. É utilizada hoje nos cartões de crédito, nos documentos de identidade (RG), no Cadastro de Pessoas Físicas (CPF), nos cheques, na correspondência postal, nos códigos de barras, nos livros (ISBN), nas publicações periódicas (ISSN), etc. Estes sistemas funcionam com variações de detalhes, tais como a forma de calcular o dígito de controle.

0.1.2 Exemplo de Aplicação: Cadastro de Pessoa Física (CPF)

O número de CPF de uma pessoa, no Brasil, é constituído de 11 dígitos, sendo um primeiro bloco com 9 algarismos e um segundo, com mais dois algarismos, que são dígitos de controle ou de verificação. A determinação desses dois dígitos de controle é feita através da congruência aritmética, por exemplo, uma relação entre dois números que, divididos por um terceiro - chamado módulo de congruência - deixam o mesmo resto. Por exemplo, o número 9 é congruente ao número 16, módulo 7, pois ambos deixam resto 2, ao serem divididos por 7. Representamos essa congruência do exemplo por $9 \equiv 16, \text{ mod. } 7$

No caso do CPF, o décimo dígito (que é o primeiro dígito verificador) é o resultado de uma congruência, módulo 11 de um número obtido por uma operação dos primeiros nove algarismos.

Se $a_1a_2a_3a_4a_5a_6a_7a_8a_9$ é a sequência formada pelos 9 primeiros dígitos do CPF, devemos multiplicá-los, nessa ordem, pela base 1,2,3,4,5,6,7,8,9 e somar os produtos obtendo um produto S . Seguindo a teoria, o primeiro dígito de controle, que vamos representar por a_{10} deve ser tal que ao ser subtraído da soma obtida, deve gerar um múltiplo de 11, isto é, se a soma obtida é S , o número $S - a_{10}$ dever ser múltiplo de 11, ou seja, $S - a_{10} \equiv 0 \text{ mod } 11$. Assim, tal número será o próprio resto da divisão por 11 da soma S obtida.

Por exemplo, se o CPF de uma pessoa tem os seguintes 9 primeiros dígitos: 235 343 104, o primeiro dígito de controle será obtido da seguinte maneira:

- Escrevemos os nove primeiros dígitos e multiplicamos cada um pela respectiva base: $2*1+3*$

$$2+5*3+3*4+4*5+3*6+1*7+0*8+4*9.$$

¹ Exercício adaptado de Professor Maurício Pereira-IC-UFGM

- Efetuando as multiplicações correspondentes, teremos:

$$2*1+3*2+5*3+3*4+4*5+3*6+1*7+0*8+4*9=116$$

- Dividindo o número 116 por 11, teremos o resultado 10 com **resto** igual a 6.
- Dessa forma, o primeiro dígito de controle será o algarismo 6.
- A determinação do segundo dígito de controle é feita de modo similar, sendo que agora acrescentamos o décimo dígito a_{10} (que é o que acabamos de calcular) e usamos uma base de multiplicação de 0 a 9. Assim teremos:

$$2*0+3*1+5*2+3*3+4*4+3*5+1*6+0*7+4*8+6*9=145$$

- Dividindo o número 145 por 11 teremos 12 com **resto** 2.
- Assim, o segundo dígito de controle é o algarismo 2.

Concluimos então que, no nosso exemplo, o CPF completo seria: 235 343 104 62. Se o resto da divisão fosse 10, usaríamos, nesse caso, o dígito zero.

0.2 Descrição do Exercício

Desenvolver um algoritmo que permita descobrir os dígitos de controle de um CPF de uma pessoa contendo apenas 9 dígitos e também permita conferir se um CPF já com os 11 dígitos está correto ou não. O sistema deve apresentar ao usuário duas opções para escolha:

1. **Calcular os dígitos de controle do CPF:** neste caso solicite os nove dígitos do CPF (você pode ler cada dígito em uma variável) – **Deve ser implementado como uma função que retorne os dígitos de controle.**
2. **Verificar a validade de um CPF:** Neste caso solicite os onze dígitos do CPF (novamente, você pode ler cada dígito em uma variável) – **Deve ser implementado como uma função que retorne se é ou não válido (True ou False).**

Seja qual for opção do usuário, deve ser apresentado na saída do sistema, o valor da soma do cálculo do primeiro dígito e a do segundo dígito. Caso a opção do usuário seja “Calcular” os dígitos, como saída exiba as informações: CPF e Dígito. Caso a opção do usuário seja “Verificar” a validade, então imprima se o CPF está correto ou não.

Teste bem o seu programa antes de entregar!