

FIAP GRADUAÇÃO

ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

Computational Thinking

PROF. EDUARDO GONDO

Agenda

- ▶ validação de CPF
- ▶ números binários
- ▶ conversão de binário para decimal
- ▶ conversão de decimal para binário
- ▶ exercícios

Algoritmo CPF

O CPF é composto por 11 dígitos sendo que os 2 últimos são chamados de dígitos de controle. Representando os dígitos do CPF por letras (ABC.DEF.GHI-JK), podemos descrever o algoritmo que gera os dígitos de controle do seguinte modo:

- ▶ primeiramente vamos gerar o dígito correspondente à letra J
- ▶ vamos multiplicar cada um dos dígitos ABCDEFGHI por 10, 9, 8, 7, 6, 5, 4, 3 e 2
- ▶ $\text{soma} = 10*A + 9*B + 8*C + 7*D + 6*E + 5*F + 4*G + 3*H + 2*I$
- ▶ a seguir pegamos o valor da soma e calculamos o resto da divisão por 11, $\text{resto} = \text{soma} \% 11$
- ▶ se o resto < 2 então o primeiro dígito verificador é 0, senão ele vale $11 - \text{resto}$

Algoritmo CPF

- ▶ agora vamos gerar o dígito correspondente à letra K
- ▶ vamos multiplicar cada um dos dígitos ABCDEFGHIJ por 11, 10, 9, 8, 7, 6, 5, 4, 3 e 2
- ▶
$$\text{soma} = 11*A + 10*B + 9*C + 8*D + 7*E + 6*F + 5*G + 4*H + 3*I + 2*J$$
- ▶ a seguir pegamos o valor da soma e calculamos o resto da divisão por 11, $\text{resto} = \text{soma} \% 11$
- ▶ se o resto < 2 então o segundo dígito verificador é 0, senão ele vale $11 - \text{resto}$

Números binários

- ▶ a nossa base de numeração é a decimal, ou seja, os dígitos para representar todos os números vão de 0 até 9
- ▶ já os computadores usam a base binária de numeração, ou seja, eles entendem apenas o 0 ou o 1 (apagado e aceso)
- ▶ em algumas situações, o computador usa também a base hexadecimal (16) de 0 a 9 e de *A* a *F*
- ▶ nos próximos slides veremos como fazer a conversão de número binário para decimal e vice-versa

Binário para decimal

- ▶ antes de aprendermos a conversão de binário para decimal, vamos trabalhar com o sistema de numeração decimal
- ▶ por exemplo considere o número 10247 que pode ser escrito como:

$$1 \cdot 10^4 + 0 \cdot 10^3 + 2 \cdot 10^2 + 4 \cdot 10^1 + 7 \cdot 10^0$$

- ▶ note que, por conta da base decimal o número pode ser decomposto como uma soma de potências de 10
- ▶ vamos agora pegar o número 10011 na base binária e convertê-lo para base decimal:

$$1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 19$$

Binário para decimal

Um outro modo de efetuar a conversão é através de uma tabela, por exemplo, considere o número 101010 na base binária

| | | | | | | |
|----|----|----|---|---|---|---|
| 64 | 32 | 16 | 8 | 4 | 2 | 1 |
| | 1 | 0 | 1 | 0 | 1 | 0 |

Agora, basta somar as potências de 2 onde há o número 1. Assim 101010 vale em decimal $32 + 8 + 2 = 42$.

Decimal para binário

- ▶ para transformarmos um número em base decimal para a base binária fazemos uma série de divisões desse número por 2
- ▶ o quociente da divisão é usado para ser o próximo número a ser dividido e o resto da divisão será parte do número na base binária
- ▶ por exemplo, vamos transformar 56 em binário:

Divisões sucessivas por 2:

$$\begin{array}{r}
 56 \div 2 = 28 \text{ resto } 0 \\
 28 \div 2 = 14 \text{ resto } 0 \\
 14 \div 2 = 7 \text{ resto } 0 \\
 7 \div 2 = 3 \text{ resto } 1 \\
 3 \div 2 = 1 \text{ resto } 1 \\
 1 \div 2 = 0 \text{ resto } 1
 \end{array}$$

E o resultado: $(56)_{10} = (111000)_2$

Exercícios CPF

1. Faça um programa que recebe um inteiro representando os números de um CPF e imprime ele formatado. Por exemplo, se o número for 12345678910, seu programa deverá imprimir 123.456.789-10.
2. Pegue seu CPF e faça o cálculo do dígito verificador usando o algoritmo apresentado. Use lápis e papel para esse propósito.
3. Desenvolva um programa em Python que calcula os dígitos verificadores de um CPF. Seu programa recebe como entrada um inteiro de 9 dígitos.
4. Usando o algoritmo anterior, desenvolva um programa que verifica se um CPF é válido, seu programa recebe um long com 11 dígitos representando um CPF e verifica se ele é válido de acordo com a regra de formação apresentada.

Exercícios Binário

1. Transforme os seguintes números de binário para decimal:
101011, 10110 e 10001
2. Transforme os seguintes números de decimal para binário:
238, 1043 e 4502
3. Escreva um algoritmo que recebe um número na base binária para a base decimal
4. Escreva um algoritmo que recebe um número na base decimal para a base binária
5. Tente fazer a conversão no papel entre números decimais para hexadecimais e vice-versa
6. Escreva um algoritmo que transforma um número inteiro na sua representação hexadecimal

Referência Bibliográfica

- ▶ Puga e Rissetti - Lógica de Programação e Estrutura de Dados
- ▶ Ascêncio e Campos - Fundamentos da Programação de Computadores
- ▶ Forbelone e Eberspacher - Lógica de programação: a construção de algoritmos e estruturas de dados
- ▶ Documentação do Python - <https://docs.python.org/3.8/>
- ▶ Python Programming For Beginners: Learn The Basics Of Python Programming (Python Crash Course, Programming for Dummies) (English Edition). Kindle
- ▶ Python: 3 Manuscripts in 1 book: - Python Programming For Beginners - Python Programming For Intermediates - Python Programming for Advanced (English Edition). Kindle

I Copyleft

Copyleft © 2022 Prof. Eduardo Gondo Todos direitos liberados.
Reprodução ou divulgação total ou parcial deste documento é liberada.