

Engenharia da Computação – 3ª série

Princípios Lógicos e Matemáticos ***(E1/E2)***

2025

Bases Numéricas:

- Conjunto de algarismos é utilizado para representar uma quantidade;
- A base numérica mais conhecida é a decimal, ou base 10, que se utiliza de 10 algarismos em sua base e suas diversas combinações, para representar qualquer valor:

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

- Na área da computação, além da base decimal (base 10), são muito utilizadas as bases binária (base 2), octal (base 8) hexadecimal (base 16), bastante focadas na construção dos circuitos dos computadores (*hardware*);

ECM306 – Tópicos Avançados em Estrutura de Dados

Princípios Lógicos e Matemáticos

- **Base Decimal (base 10):**

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

- **Base Binária (base 2):**

0, 1

- **Base Octal (base 8):**

0, 1, 2, 3, 4, 5, 6, 7

- **Base Hexadecimal (base 16):**

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F

ECM306 – Tópicos Avançados em Estrutura de Dados

Princípios Lógicos e Matemáticos

Exemplos:

- Base Decimal (base 10):

1320_{10} ou 1320

- Base Binária (base 2):

11111010_2

- Base Octal (base 8):

1705_8

- Base Hexadecimal (base 16):

$1F3B_{16}$

Curiosidades:

- A linguagem de máquina é a compreendida e executada internamente pelos computadores atuais, em seus circuitos eletrônicos;
- Na linguagem de máquina, tudo é representado pela base binária, com suas sequências de 0's e 1's, somente;
- Normalmente, o 0 (zero) representa um circuito eletrônico desligado (sem corrente elétrica) e o 1 (um) representa um circuito eletrônico ligado (com corrente elétrica), daí serem chamados de circuitos eletrônicos digitais (um dígito 0 ou 1);
- Num circuito eletrônico digital, cada dígito representado por 0 (zero) ou por 1 (um) é denominado de *bit* (binary digit).

Notação Posicional:

- Os sistemas de numeração utilizam a notação posicional para definir o valor de cada número para ser convertido para a base decimal (base 10):

$$1001_2 = 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 9_{10}$$

$$217_8 = 2 \times 8^2 + 1 \times 8^1 + 7 \times 8^0 = 143_{10}$$

$$F1CA_{16} = 15 \times 16^3 + 1 \times 16^2 + 12 \times 16^1 + 10 \times 16^0 = 61898_{10}$$

$$123_{10} = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 = 123_{10}$$

Método das Divisões Consecutivas:

- Os sistemas de numeração utilizam o método das divisões consecutivas definir o valor de cada número para ser convertido da base (base 10):

$$11_{10} = (?)_2$$

$$11 \mid \underline{2}$$

$$\textcolor{red}{1} \ 5 \mid \underline{2}$$

$$\textcolor{red}{1} \ 2 \mid \underline{2}$$

$$\textcolor{red}{0} \ 1 \mid \underline{2}$$

$$\textcolor{red}{1} \ \underline{0}$$

$$= \textcolor{red}{1011}_2$$

Método das Divisões Consecutivas:

- Encontre o valor equivalente na base 10:

$$9_{10} = (?)_2$$

$$143_{10} = (?)_8$$

$$61898_{10} = (?)_{16}$$

$$123_{10} = (?)_{10}$$

Lógica Booleana:

- Como já sabido, o computador utiliza o sistema binário para trabalhar com os dados;
- Além das operações aritméticas, o computador executa as operações lógicas, que são “E”, “OU” e “NÃO”;
- Para realizar tais operações lógicas, o computador faz uso de circuitos lógicos, chamados de portas lógicas;
- A partir dessas portas lógicas, podem ser feitas diversas combinações, utilizando o conceito chamado de Álgebra de Boole;
- Essas combinação são representadas por expressões, chamadas de Expressões Booleanas.

Lógica Booleana:

- A lógica utilizada na Álgebra de Boole é chamada de Lógica Booleana;
- Essa lógica considera apenas 2 valores: 0 (zero) e 1 (um);
- Portanto, a Lógica Booleana utiliza a base binária;
- Na Lógica Booleana, o 0 (zero) representa o “não”/“falso” e o 1 (um) representa o “sim”/“verdadeiro”;
- Cada porta lógica representa uma operação lógica que, para cada combinação de entradas, retorna uma saída diferente.

Tabela da Verdade:

- Portanto, existe uma tabela que permite mapear as possíveis entradas e saídas das expressões booleanas, denominada Tabela da Verdade;
- A Tabela da Verdade lista as possíveis combinações dos valores lógicos de entrada (0 ou 1) e mapeia a saída de cada uma delas.

Tabela da Verdade: Lógica E (AND ou &):

- A operação “E” é uma operação binária, ou seja, precisa de dois operandos:

<u>A</u>	<u>B</u>	<u>A & B</u>
0	0	0
0	1	0
1	0	0
1	1	1

Onde: 0 é F/f (Falso/false);
 1 é V/t (Verdadeiro/true).

Tabela da Verdade: Lógica OU (OR ou |):

- A operação “OU” é uma operação binária, ou seja, precisa de dois operandos:

<u>A</u>	<u>B</u>	<u>A B</u>
0	0	0
0	1	1
1	0	1
1	1	1

Onde: 0 é F/f (Falso/false);
 1 é V/t (Verdadeiro/true).

Tabela da Verdade: Lógica NÃO (NO ou \sim):

- A operação “NÃO” é uma operação unária, ou seja, precisa de um operando:

<u>A</u>	<u>$\sim A$</u>
0	1
1	0

Onde: 0 é F/f (Falso/false);
 1 é V/t (Verdadeiro/true).

ECM306 – Tópicos Avançados em Estrutura de Dados

Aula Introdutória

FIM