

Trabalho Prático 1

Objetivos

- Representação da informação
- Notação posicional
- Conversão entre bases
- Operações aritméticas básicas em várias bases
- Representação de quantidades negativas

Introdução

A informação numérica pode assumir diversos formatos, desde a representação romana (e.g., XXVII) até à representação decimal (e.g., 27_{10}). A diferença entre estes dois formatos consiste no facto de no primeiro caso o peso de cada algarismo (e.g., X) não depender da posição que ocupa, ao passo que no sistema decimal cada algarismo tem um 'peso' associado à posição, ou seja, no caso anterior o algarismo '2' tem um peso de 10 (10^1) e o algarismo '7' tem um peso de 1 (10^0), sendo 10 a base de representação. Os sistemas que iremos abordar são todos sistemas de representação posicional com base decimal, hexadecimal, octal e binária.

Nos sistemas posicionais o número N , na base r , é representado por $(d_{k-1} d_{k-2} \dots d_1 d_0)_r$ onde o dígito da esquerda tem um peso maior. O respetivo valor decimal de N é obtido através da ponderação dos dígitos, através da expressão seguinte:

$$(\text{Valor de } N)_{10} = d_{k-1} \times r^{k-1} + d_{k-2} \times r^{k-2} + \dots + d_1 \times r + d_0$$

Guião

1. Representação e conversão entre bases (sem sinal)

1.1 Represente no sistema decimal o valor das seguintes quantidades:

- | | | |
|-------------------|-------------------|--------------------|
| a) 1010111001_2 | b) $DF6_{16}$ | c) 10110111101_2 |
| d) $A7A2_{16}$ | e) 1111111111_2 | f) $40F0_{16}$ |
| g) 2024_8 | | |

1.2 Represente nos sistemas hexadecimal e binário o valor das seguintes quantidades:

- | | | |
|------------------|-----------------|----------------|
| a) 1025_{10} | b) 33427_{10} | c) 7543_{10} |
| d) 110110111_2 | e) $DAD0_{16}$ | f) 7254_8 |

2. Aritmética binária, octal e hexadecimal (sem sinal)

2.1 Calcule o resultado da soma dos seguintes pares de valores:

- a) $10101101_2 + 11100001_2$ b) $1011011_2 + 111110_2$
c) $125_{16} + 1A7_{16}$ d) $111011_2 + 107_8$

2.2 Calcule o resultado da subtração dos seguintes pares de valores:

- a) $11100001_2 - 10101101_2$ b) $1011011_2 - 1001001_2$
c) $30A_{16} - 2FF_{16}$ d) $135_{16} - 135_8$

2.3 Calcule o resultado da multiplicação dos seguintes pares de valores:

- a) $11100001_2 * 10001101_2$ b) $25_{16} * 17_{16}$
c) $3CA_{16} * 202_{16}$ d) $777_8 * 10111_2$

3. Inteiros com sinal (Complemento a 2)

Assumindo que as quantidades seguintes estão codificadas em complemento para 2, com 8 bits de representação, indique o seu equivalente decimal:

- a) 11111110_2 b) 00000000_2 c) 11111111_2
d) 00110011_2 e) 11001100_2 f) 10001110_2

4. Complemento para 2 e número de bits de representação

4.1 Indique a representação (quando possível) das quantidades seguintes quando codificadas em complemento para 2 e armazenadas num registo de 12 bits:

- a) -127_{10} b) $+145_{10}$ c) $-5F6_{16}$
d) -01100_2 e) -2045_{10} f) $+ABC_{16}$

4.2 Assumindo que as quantidades seguintes estão codificadas em complemento para 2, com 8 bits de representação, determine, sempre que for possível, a representação das mesmas quantidades em complemento para 2 com 5 bits:

- a) 11110101_2 b) 00001010_2 c) 11001100_2
d) 11111110_2 e) 10111111_2 f) 11110000_2

5. *Overflow* nas operações aritméticas binárias de soma e subtração

Calcule o resultado das operações seguintes em complemento para 2, com 8 bits de representação. Comece por determinar a representação de cada um dos operandos. Identifique os casos em que ocorre *overflow*.

- | | | |
|------------------------------|------------------------------|----------------------------|
| a) $-1_{10} + 63_{10}$ | b) $123_{10} + 46_{10}$ | c) $12_{10} + (-124_{10})$ |
| d) $-125_{10} + (-128_{10})$ | e) $11111100_2 - 11100101_2$ | f) $-10_{16} + (-01100_2)$ |

6. Operações lógicas binárias: OR, AND, XOR, NOR, NAND e XNOR

Determine o resultado das operações seguintes:

- | | |
|-------------------------------------|--------------------------------------|
| a) $11110000 \text{ OR } 10101011$ | b) $11110000 \text{ AND } 10101011$ |
| c) $11110000 \text{ XOR } 10101011$ | d) $11110000 \text{ NAND } 10101011$ |
| e) $11110000 \text{ NOR } 10101011$ | f) $11110000 \text{ XNOR } 10101011$ |

7. Exercícios adicionais

7.1 Exprima nos sistemas decimal e binário o valor da maior quantidade inteira não negativa que pode ser representada num registo com capacidade de armazenamento de 4 símbolos hexadecimais.

7.2 Represente no sistema decimal, tendo o cuidado de manter aproximadamente a precisão da representação original, o valor das quantidades racionais não negativas seguintes:

- | | | | |
|-----------------------|-----------------|--------------------|----------------|
| a) 101110.1100101_2 | b) $2B4.0_{16}$ | c) 111000.1010_2 | d) $2F.4_{16}$ |
|-----------------------|-----------------|--------------------|----------------|

7.3 Represente nos sistemas hexadecimal e binário, tendo o cuidado de manter aproximadamente a precisão da representação original, o valor das quantidades racionais não negativas seguintes:

- | | | |
|-----------------|------------------|-----------------|
| a) 10.25_{10} | b) 33.427_{10} | c) 754.3_{10} |
|-----------------|------------------|-----------------|