

**Universidade Federal de Campina Grande Centro de
Engenharia Elétrica e Informática Unidade
Acadêmica de Sistemas e Computação
Curso de Bacharelado em Ciência da Computação**

Introdução à Computação

A Informação e sua Representação (Parte III)

Prof.a Joseana Macêdo Fechine Régis de Araújo
joseana@computacao.ufcg.edu.br

Carga Horária: 60 horas



A Informação e sua Representação

Representação de algarismos

❑ Linguagem humana

- ❑ O sinal (= ou -) é um símbolo de forma diversa daquela utilizada para os algarismos que representam o valor do número (0, 1, 2, ..5, ...).
- ❑ O sinal é colocado separado da magnitude (+31), não fazendo parte dos cálculos em si, mas tão-somente servindo para **definir o sinal do resultado e o tipo da operação a ser realizada efetivamente.**

❑ Sistemas de computação

- ❑ O sinal é um símbolo de forma idêntica à dos algarismos representativos do seu valor (é um bit igual aos demais).
- ❑ O bit de sinal está incorporado aos bits da magnitude, formando um único número.



Operações aritméticas – Sistema Binário

- ❑ Adição
- ❑ Subtração
- ❑ Multiplicação
- ❑ Divisão

As operações aritméticas nos sistemas binário, octal, decimal e hexadecimal obedecem a regras similares.



Adição – Sistema binário

- ❑ **Adição** → deslocamento à direita na série, cada deslocamento correspondendo a adição de uma unidade.
- ❑ Adição entre dois números de um algarismo → pode-se obter resultados com um ou dois dígitos.
- ❑ **Estouro** (maior algarismo é ultrapassado): **Carry** (transporte) ou **vai-um**.



Adição – Sistema binário

Regra:

$$\begin{array}{r} \text{Carry} \leftarrow \\ A \\ B + \\ \hline S \end{array}$$

Operandos		Resultado	Estouro
A	B	S	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



Subtração – Sistema binário

- ❑ Inversa à adição → **deslocamento à esquerda do minuendo** de tantas unidades quantas forem o **subtraendo**.
- ❑ Se o minuendo é menor que o subtraendo ?
Estouro → **subtrair** uma unidade do minuendo ou **somar** uma unidade ao subtraendo da casa seguinte.
- ❑ Estouro → **borrow** (empréstimo) ou **vem-um**.



Subtração – Sistema binário

Regra:

$$\begin{array}{r} A \\ B - \\ \hline S \end{array}$$

Borrow ←

Operandos		Resultado	Estouro
A	B	S	Borrow
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0



Outras Operações Aritméticas

Multiplicação – Sistema binário

Regra

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

Multiplicação pela base

- ❑ Desloca-se os **algarismos** de um número para a **esquerda** ou a sua **vírgula** para a **direita**.



Outras Operações Aritméticas

Divisão – Sistema binário

- ❑ Procedimento igual ao dos decimais, considerando-se apenas que: $0/1 = 0$ e $1/1 = 1$.

↘ Divisão pela Base

- ❑ Deslocando-se os **algarismos** de um número para a **direita** ou a sua vírgula para a esquerda.



A Informação e sua Representação

- ❑ **Aritmética em Sinal e Magnitude (adicional)**
- ❑ **Aritmética em Complemento de 2**
- ❑ **Aritmética em Ponto Flutuante (adicional)**
- ❑ **Aritmética em BCD (adicional)**



Aritmética em Complemento de 2

Algoritmo para operação aritmética de adição

1. Somar os dois números, bit a bit, inclusive o bit de sinal.
2. Desprezar o último “vai 1” (para fora do número), se houver.
3. Se, simultaneamente, ocorrer “vai 1” para o bit de sinal e “vai 1” para fora do número, ou se ambos não ocorrerem, o resultado está correto.
4. Se ocorrer apenas um dos dois “vai 1” (ou para o bit de sinal ou para fora), o resultado está incorreto. Ocorreu um **overflow**.
 - ❑ O **overflow** somente pode ocorrer se ambos os números tiverem o mesmo sinal (seja positivo ou ambos negativos) e, nesse caso, se o sinal do resultado for oposto ao dos números.



Aritmética em Complemento de 2

Algoritmo para operação aritmética de subtração

1. Complementar a 2 o subtraendo, independentemente se é um valor positivo ou negativo.
2. Somar os números, utilizando o algoritmo da adição já mostrado anteriormente.



Aritmética em Complemento de 2

Resumindo, é importante lembrar que:

- ❑ As operações de adição e subtração são normalmente realizadas como adição.
- ❑ As operações de subtração são realizadas como soma de complemento (minuendo mais o complemento do subtraendo).
- ❑ Se o resultado encontrado é um **valor positivo**:
 - ❑ o valor decimal correspondente da magnitude é obtido por pura conversão de base 2 para base 10.
- ❑ Se o resultado encontrado é um **valor negativo**:
 - ❑ deve-se primeiro converter esse valor para representação de sinal e magnitude (consistirá em trocar o valor dos bits da magnitude e somar 1 ao resultado) e, em seguida, converter a magnitude de base 2 para base 10.



Aritmética em Complemento de 2

□ **Exemplo:** Realize as operações aritméticas a seguir (em complemento de 2). Considere a palavra de dados com 6 bits.

a) $(+13)_{10} + (+15)_{10}$

b) $(+23)_{10} + (+20)_{10}$

c) $(+15)_{10} + (-13)_{10}$

d) $(+20)_{10} - (+17)_{10}$

e) $(-24)_{10} - (-15)_{10}$

f) $(-24)_{10} - (+15)_{10}$



Aritmética em Complemento de 2

a) $(+13)_{10} + (+15)_{10}$

	001111
+13	001101
+15	001111
<hr/>	
+28	011100

Resultado correto - não houve “vai 1” nem para o bit de sinal nem para fora do número.

b) $(+23)_{10} + (+20)_{10}$

	010100
+23	010111
+20	010100
<hr/>	
+43	101011

Resultado incorreto - houve “vai 1” apenas para o bit de sinal.

Overflow - faixa de representação para 6 bits (-32 a 31)



Aritmética em Complemento de 2

c) $(+15)_{10} + (-13)_{10}$

	111111
+15	001111
-13	110011
<hr/>	
+2	000010

d) $(+20)_{10} - (+17)_{10}$

	111100
+20	010100
-17	101111
<hr/>	
+3	000011

Resultados corretos - houve “vai 1” para o bit de sinal e para fora do número; este é desprezado.



Aritmética em Complemento de 2

e) $(-24)_{10} - (-15)_{10}$

	001000
-24	101000
+15	001111
<hr/>	
-9	110111

Resultado correto - não houve “vai 1” para o bit de sinal nem para fora do número.

f) $(-24)_{10} - (+15)_{10}$

	100000
-24	101000
-15	110001
<hr/>	
-39	011001

Resultado incorreto - houve “vai 1” apenas para fora do número.

Overflow - faixa de representação para 6 bits (-32 a 31)



Aritmética em Complemento de 2

- ❑ A aritmética em complemento de 2 **requer apenas um componente (somador)** para somar dois números e **um componente** que realize a operação de **complementação**.
- ❑ O algoritmo básico refere-se, então, à soma dos números, considerando-se que os números negativos estejam representados em complemento de 2; ele acusa, também, se o resultado ultrapassar a quantidade de bits representáveis na ULA, **overflow**.

Pode-se efetuar a multiplicação através de sucessivas somas e a divisão a partir de sucessivas subtrações (processo lento!).



Operações aritméticas – Sistema Binário

Observação:

- ❑ A multiplicação em computadores pode ser feita por um artifício:
 - ❑ para multiplicar um número **A** por **n**, basta somar **A** com **A**, **n** vezes.
 - ❑ Por exemplo, $4 \times 3 = 4 + 4 + 4$.
- ❑ A divisão também pode ser feita por subtrações sucessivas!
- ❑ **O que concluir? Uma operação aritmética pode ser realizada em computadores apenas por meio de somas (diretas ou em complemento)!**



A Informação e sua Representação

- ❑ A diferença de formas de representação e respectivos algoritmos de realização de operações matemáticas é bastante útil, pois cada uma tem uma aplicação na qual é mais vantajosa que a outra.
- ❑ Cabe ao programador a escolha da forma a ser utilizada pelo sistema, podendo ser:
 - ❑ **explícita** - o programador define as variáveis e constantes em seu programa.
 - ❑ **implícita** - o programador deixa que o compilador faça sua própria escolha.



A Informação e sua Representação

- ❑ Qualquer dos métodos utilizados para representar números em computador (sinal e magnitude, complemento de 2, ponto flutuante, BCD) requer atenção do programador pelo fato de que algumas operações realizadas podem redundar em erro no resultado. **Motivo:** quantidade finita e limitada de algarismos utilizada nas máquinas.
- ❑ Números inteiros são menos problemáticos - o programador pode criar uma faixa menor e mais definida de valores em seu programa, porém, com números reais (fracionários), o problema se torna muito maior à medida que é possível criar infinitos valores entre qualquer faixa de valores.



A Informação e sua Representação

- ❑ Diferentemente de nossos cálculos usando papel e lápis, cujo único limite é o tamanho do papel, em computação precisa-se ter atenção aos limites impostos pela quantidade máxima de bits dos valores representados e dos diversos componentes da máquina (registradores, barramento, etc.) - **Tais limites afetam a precisão dos resultados.**



A Informação e sua Representação

Quadro resumo dos tipos primitivos de dados
(entendidos pelo hardware do computador)

❑ **Caractere**

❑ **Numérico**

❑ **Lógico**

Ponto Fixo

Ponto Flutuante

Decimal

Sinal e magnitude

Complemento de 2

Permite a utilização de variáveis que possuem apenas dois valores para representação, FALSO (0) e VERDADEIRO (1). Permite realizar também operações que empregam operadores lógicos encontrados nos computadores. Para o melhor entendimento é necessário o estudo de **conceitos da Lógica Digital**.