



# Os Computadores

## A Informação nos Computadores



# A Informação nos Computadores

- ❑ Nós conseguimos guardar informações sob a forma de imagens e cenas
- ❑ Entretanto, o computador é uma máquina, sendo muito difícil construir circuitos para que ele guarde imagens, tal como o cérebro
- ❑ Então como é que as informações são *manipuladas* pelo computador?

# A Informação nos Computadores

## □ A maneira mais fácil:

presença/ausência de sinal elétrico



Ligado/desligado  
aceso/apagado  
1/0



Como tem-se dois estados,  
chama-se essa representação de  
**Codificação Binária**

# A Informação nos Computadores

## □ A maneira mais fácil:

Exemplo: O número 2 em binário equivale a



1

0

... O que pode ser representado por





# A Informação nos Computadores

- ❑ A menor unidade que transita em um computador é o **BIT**
  - ↙ que representa os estados aceso/apagado, ligado/desligado, 1/0
- ❑ Um símbolo, seja ele uma letra, um número, um sinal, é representado por um **conjunto de bits**
  - ↙ A esse conjunto de Bits denomina-se **BYTE**
- ❑ Tipicamente, 1 (um) **Byte** tem 8 (oito) **Bits**



# A Informação nos Computadores

- ❑ Por esse motivo, alguns componentes do computador são dimensionados em **Bits ou Bytes**
- ❑ Como cada **símbolo** precisa de 1 byte para ser representado, um **texto** precisará de vários bytes



# A Informação nos Computadores

- ❑ Qualquer texto ou informação serão guardados no computador sob a forma de  
**Arquivos de Dados**
- ❑ Como o sistema de codificação é binário, as quantidades que representam grupos de bytes armazenados ou transportados no computador são potências de 2.



# A Informação nos Computadores

## □ Assim:

↙ 1 KByte = 1.024 *Bytes*

↙ 1 MByte = 1.024 *KBytes* ou  
1.048.576 *Bytes*

↙ 1 GByte = 1.024 *MBytes* ou  
1.048.576 *KBytes* ou  
1.073.741.824 *Bytes*

↙ KByte = KiloByte = KB = K

↙ MByte = MegaByte = MB = Mega

↙ GByte = GigaByte = GB = Giga





# Representação de Dados

- Os dados e as informações em um computador são representados internamente das seguintes formas:





# Representação de Dados

## □ Representação Alfanumérica

### ↳ Representação ASCII

- ☐ originalmente utilizava 7 bits, suficientes para representar 128 caracteres
- ☐ só serve para língua inglesa, pois não inclui acentos e símbolos utilizados em alguns idiomas
- ☐ atualmente utiliza 8 bits, mas o problema da representação limitada continua



# Representação de Dados

## □ Representação Alfanumérica

### ↳ Representação EBCDIC

- ▣ assim como ASCII, também utilizava 7 bits inicialmente e hoje utiliza 8 bits
- ▣ usado
  - nos mainframes IBM
  - computadores Macintosh (Apple)
- ▣ apesar de utilizar o mesmo número de bits que o ASCII, a representação de um símbolo difere nos dois códigos




# Representação de Dados

## □ Representação Alfanumérica

### ↙ Representação UNICODE

- ☰ os códigos anteriores são insuficientes para representar símbolos de outros idiomas além do inglês
- ☰ este código amplia a representação para 16 bits, o que possibilita mais de 65.000 símbolos
- ☰ adotado por
  - Apple, HP, IBM, Microsoft, Oracle, Sun, Sybase, Unisys



<i>Caractere</i>	<i>Representação ASCII</i>	<i>Representação EBCDIC</i>
0	00110000	11110000
1	00110001	11110001
2	00110010	11110010
3	00110011	11110011
4	00110100	11110100
5	00110101	11110101
6	00110110	11110110
7	00110111	11110111
8	00111000	11111000
9	00111001	11111001
A	01000001	11000001
B	01000010	11000010
C	01000011	11000011
D	01000100	11000100
E	01000101	11000101
F	01000110	11000110
G	01000111	11000111
H	01001000	11001000
I	01001001	11001001
J	01001010	11010001
K	01001011	11010010
L	01001100	11010011
M	01001101	11010100
N	01001110	11010101
O	01001111	11010110
P	01010000	11010111
Q	01010001	11011000
R	01010010	11011001
S	01010011	11100010
T	01010100	11100011
U	01010101	11100100
V	01010110	11100101
W	01010111	11100110
X	01011000	11100111
Y	01011001	11101000
Z	01011010	11101001



# Representação de Dados

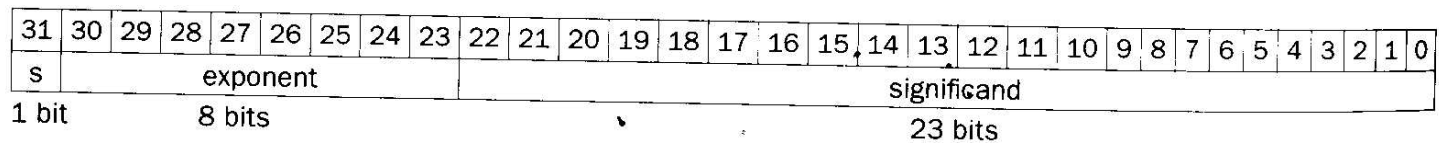
## □ Representação Numérica

- ↙ Os códigos de caracteres não servem para representar números fracionários
- ↙ para esses números e para números muito grandes é usada a notação de ponto flutuante
  - ☰ os números são representados como potências de 2:  
$$(-1)^S \times F \times 2^E$$
, onde S = sinal  
F = valor ponto flutuante  
E = expoente
  - ☰ Precisão simples: 32 bits
  - ☰ Precisão dupla: 64 bits

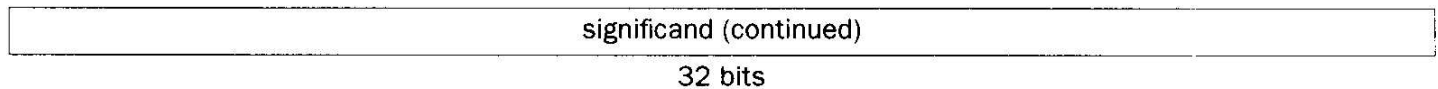
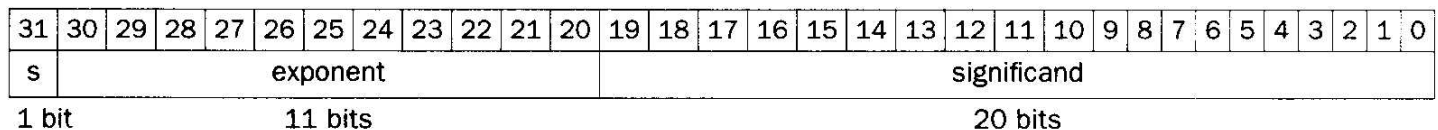
# Representação de Dados

## □ Representação Numérica

### ↙ Padrão IEEE 754



simples



dupla



# Representação de Dados

## ❑ Outras bases de representações

- ↙ Algumas vezes são utilizadas outras bases de representação além da binária para facilitar a escrita

- ↙ Bases mais usadas

  - ☐ Hexadecimal – base 16

  - ☐ Octal – base 8

- ↙ Isso não quer dizer que o computador represente internamente nessas bases!

  - ☐ Elas são usadas por programadores