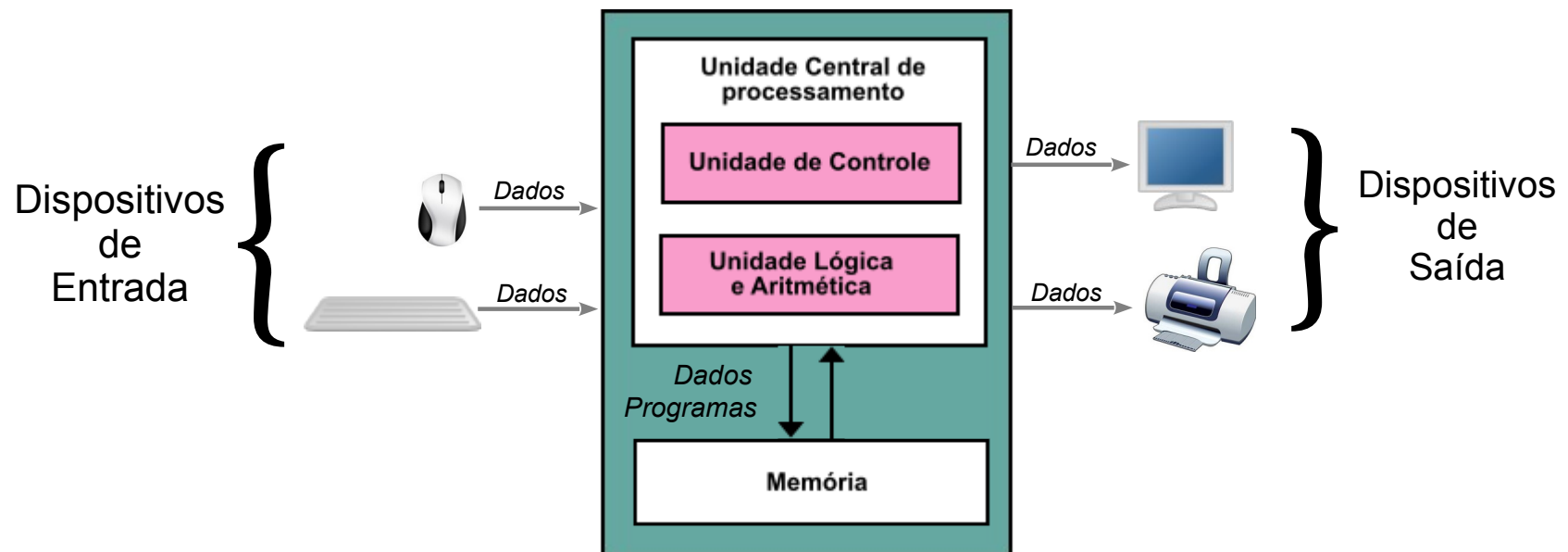


# Computadores, Dados e Programas

# Computadores

- Computador
  - Dispositivo eletrônico que armazena e processa **dados** (números, sons, textos, imagens, etc.)
- Modelo de Von Neumann



Fonte: By Kapooh - Own work, CC BY-SA 3.0

# Computadores e Dados

- Dados são representados por ***estados binários*** (bits)
  - ▶ 1 (Verdadeiro)
  - ▶ 0 (Falso)
- Como números, palavras, imagens, sons e vídeos podem ser representados por 0's e 1's?

# Números e contas

- 10 dedos → sistema de numeração **decimal** (base 10)
- Se tivéssemos 2 dedos → sistema de numeração **binária** (base 2)

	Valor Decimal	Valor Binário	
	0	0	
$1 \times 10^0$	1	1	$1 \times 2^0 = 1$
$2 \times 10^0$	2	10	$1 \times 2^1 + 0 \times 2^0 = 2$
$3 \times 10^0$	3	11	$1 \times 2^1 + 1 \times 2^0 = 3$
	4	100	
$5 \times 10^0$	5	101	$1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5$
	8	110	
	9	111	
	15	1111	
$2 \times 10^1 + 0 \times 10^0$	20	10100	$1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 = 20$
$1 \times 10^2 + 3 \times 10^1 + 5 \times 10^0$	130	1000 0010	← Quanto maior o valor → mais bits

# Letras e símbolos

- **Tabela ASCII** (1963): código binário que codifica 128 símbolos
- **Ideia:** 0's e 1's enviados ao monitor, são “traduzidas” visualmente nos símbolos que conhecemos
- Uma parte da tabela:

Binário	Símbolo
0010 1101	-
0010 1110	.
0010 1111	/
0011 0000	0
0011 0001	1
0011 0010	2
0011 0011	3

Binário	Símbolo
0100 1101	M
0100 1110	N
0100 1111	O
0101 0000	P
0101 0001	Q
0101 0010	R
0101 0011	S

Binário	Símbolo
0110 1101	m
0110 1110	n
0110 1111	o
0111 0000	p
0111 0001	q
0111 0010	r
0111 0011	s

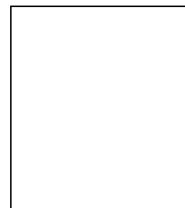
# Imagens

- **Como representar uma imagem com 1 único bit?**

**se bit é 0 então...**



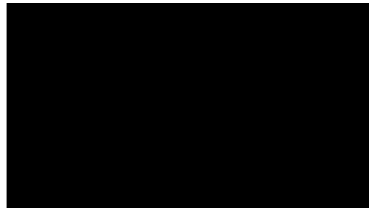
**se bit é 1 então...**



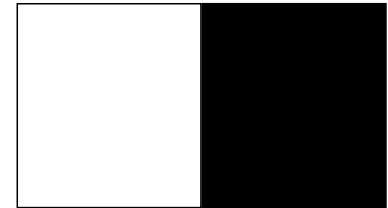
# Imagens

- **Com 2 bits então?**

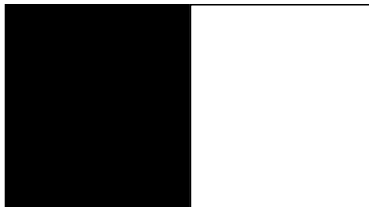
se 00 então...



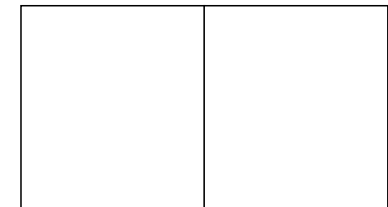
se 10 então...



se 01 então...



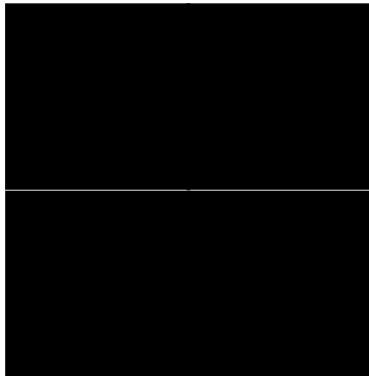
se 11 então...



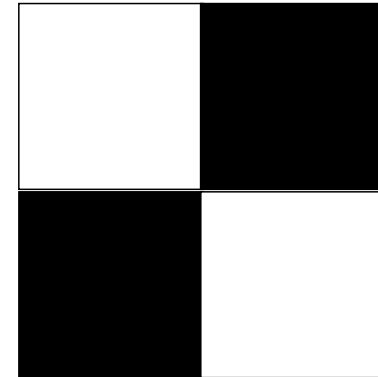
# Imagens

- O que se pode fazer com 4 bits?

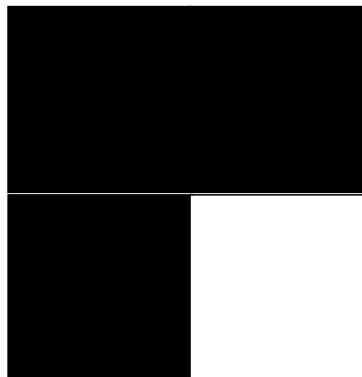
00  
00



10  
01



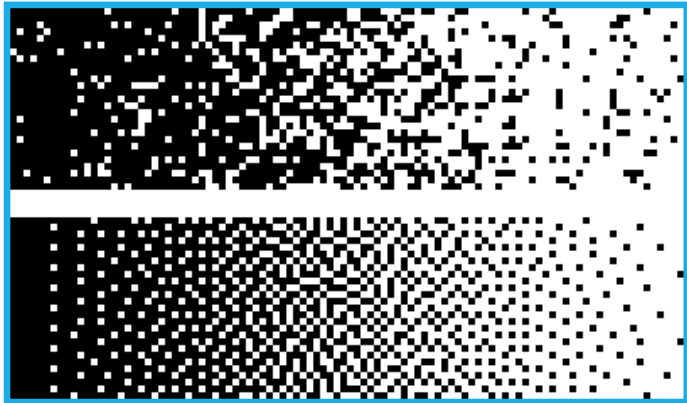
00  
01





# Imagens















- O que posso fazer com **VÁRIOS** bits?



A imagem é uma **MATRIZ** de pontos !

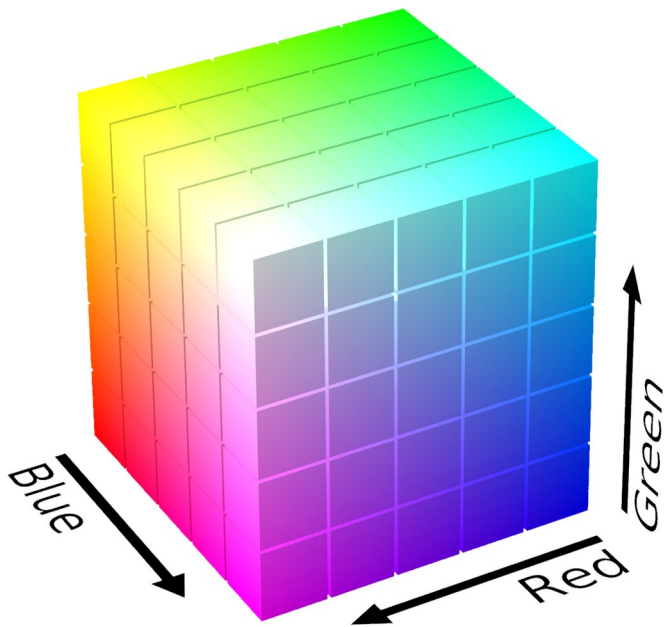
# Imagens

- **E as cores?** → Tons de cinza para cada ponto

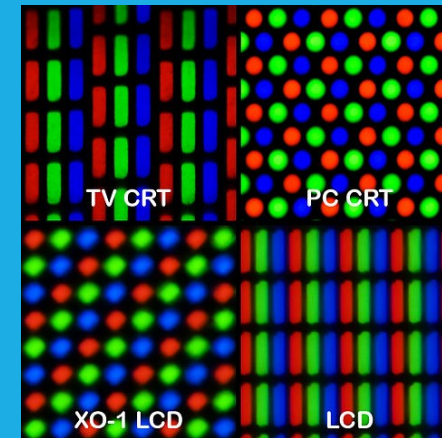
1 bit		2 bits		3 bits	
0		00		000	
1		01		001	
		10		010	
		11		011	
				100	
				101	
				110	
				111	

# Imagens

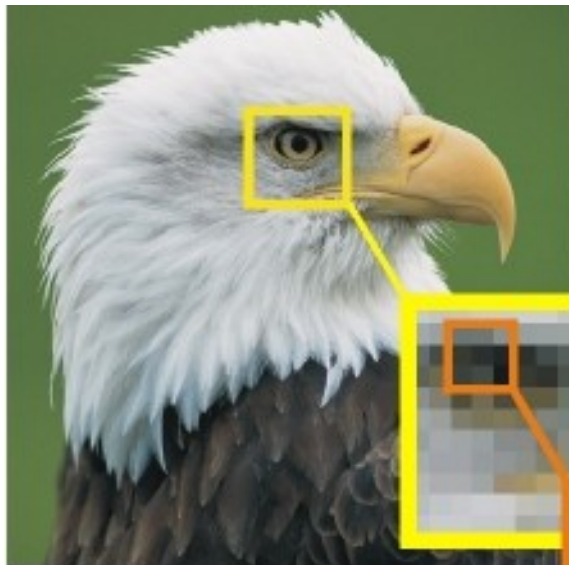
- Mas, e as cores? → Sistema **RGB**



Não é a tãa que a TV vista BEM de perto é assim:



# Imagens



Um pixel

**Imagem:** matriz de pixels

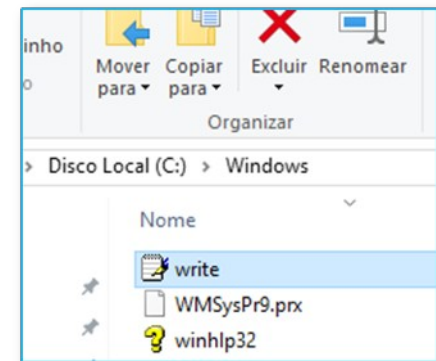
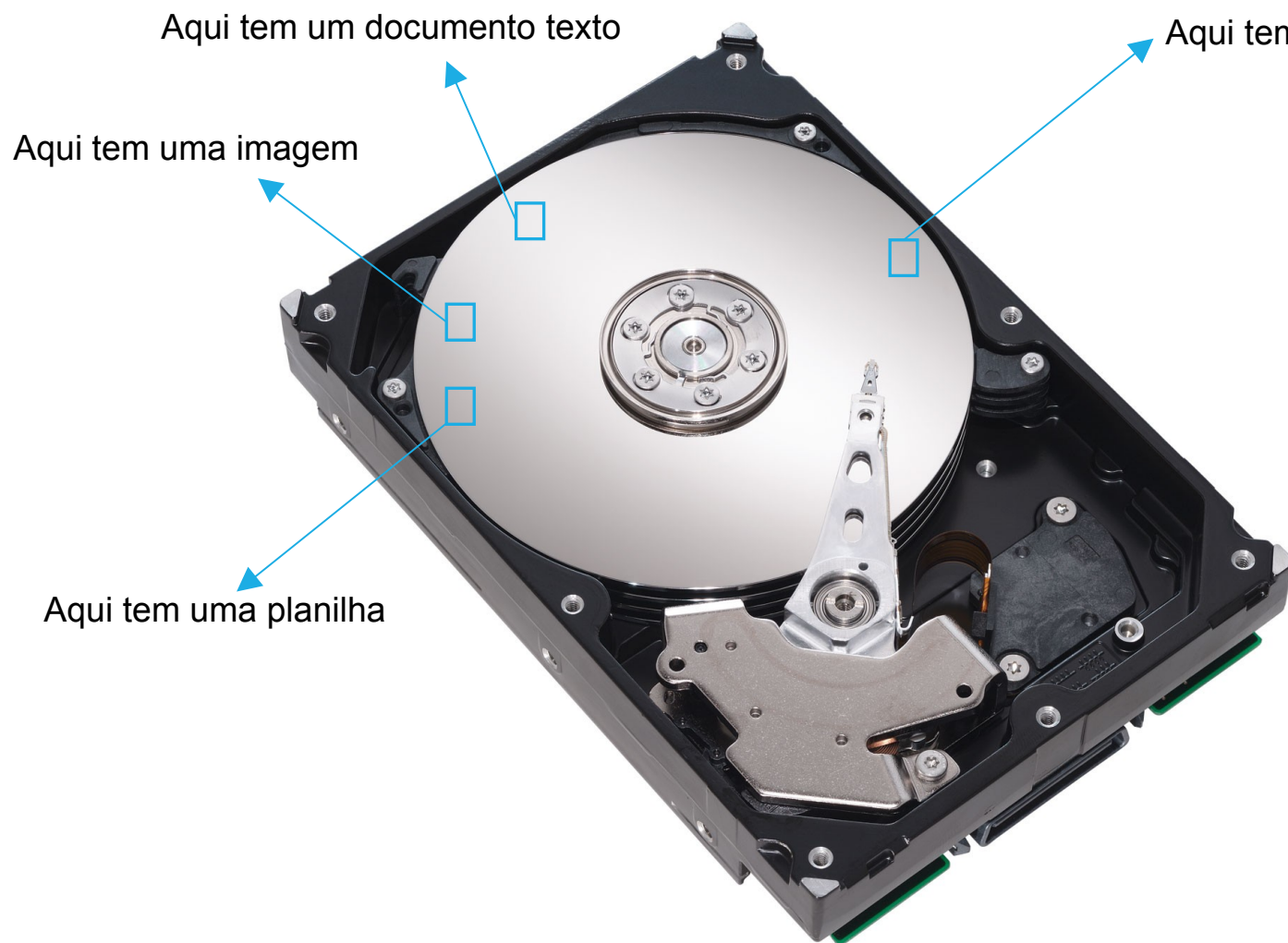
**Pixel:**

- ponto da imagem
- 24 bits

1001  
0110  
0011  
1101  
0111  
0011

8 bits para VERMELHO  
8 bits para VERDE  
8 bits para AZUL

# Disco Rígido (HD)



# Programas e Algoritmos

- Se **dados** são representados por bits, como representar o **PROCESSAMENTO** destes dados?
  - ▶ Sequência de instruções executadas no computador
  - ▶ Cada instrução é representada no computador por **bits**
    - ▷ Detalhes são vistos em cursos de Ciência da Computação (normalmente)
- O que é um **Programa**?
  - ▶ Sequência de instruções executadas no computador
  - ▶ Codificação de um **algoritmo**

# Próximos Passos

- O que é um **Algoritmo**?
  - ▶ Veremos na sequência ...

**Créditos:** O conteúdo original deste documento é de autoria do Prof. André Vingnatti (UFPR/DINF), e foi alterado pelo Prof. Armando Delgado (UFPR/DINF) para uso na disciplina *Programação de Computadores* (CI208, CI180, CI183)

Compartilhe este documento de acordo com a licença abaixo



Este documento está licenciado com uma Licença *Creative Commons Atribuição-NãoComercial-SemDerivações* 4.0 Internacional.  
<https://creativecommons.org/licenses/by-nc-sa/4.0/>