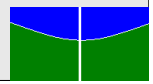
ALGORITMOS E PROGRAMAÇÃO DE COMPUTADORES

Disciplina: 113476

Profa. Carla Denise Castanho

Universidade de Brasília – UnB
Instituto de Ciências Exatas – IE
Departamento de Ciência da Computação – CIC

2. ORGANIZAÇÃO BÁSICA DE UM COMPUTADOR

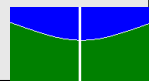


Hardware & Software

- Hardware:
  - Corresponde à parte material, os componentes físicos do sistema; é o computador propriamente dito. Incluindo periféricos de entrada e saída; a máquina, seus elementos físicos, carcaça, placas, fios, fonte e componentes em geral.
  - Um hardware sozinho não é nada, a menos que ele tenha uma função a executar e um programa que lhe diga como executá-la.
    - ***"É a parte que vc xinga!"***
- Software:
  - São instruções escritas em linguagem de programação que dirão ao computador o que fazer e auxiliarão o usuário em suas atividades. Ou seja, os programas e os sistemas de programação utilizados por um computador e que permitem atender às necessidades do usuário.
    - ***"É a parte que vc chuta!"***

Principais Classes de Sistemas Computacionais

- Servidores
- Pessoais
- Embarcados



Principais Classes de Sistemas Computacionais

- Servidores
  - Recursos compartilhados entre vários usuários
  - Geralmente sistemas de software específicos
  - Ex.: Desde simples servidores de arquivo, webservers até supercomputadores
  - Alta dependabilidade (confiabilidade, segurança, disponibilidade e mantenabilidade), geralmente alto custo.
- Pessoais
- Embarcados

Tianhe-2

Face Book

Principais Classes de Sistemas Computacionais

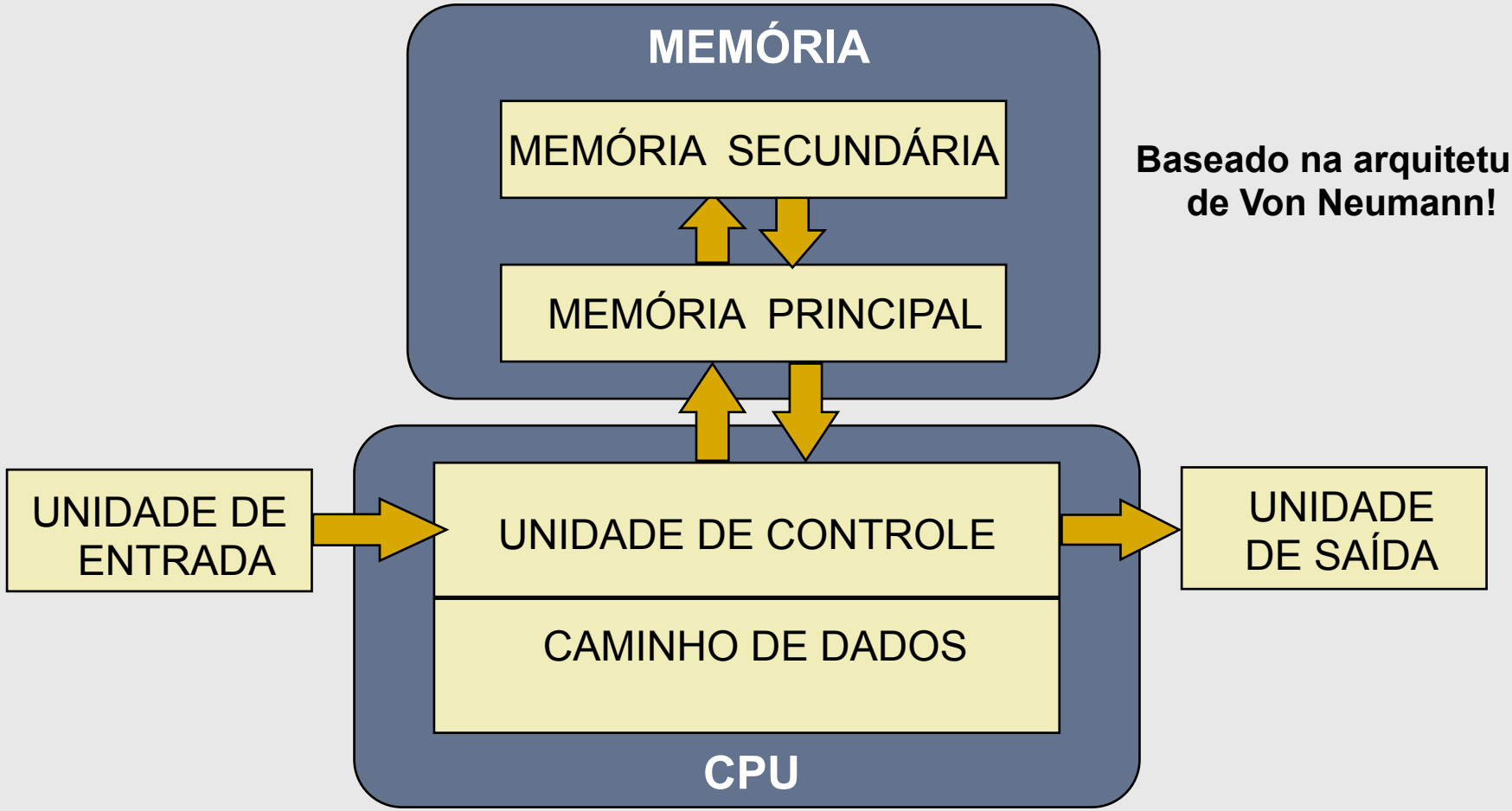
- Servidores
- Pessoais
  - Recursos utilizados geralmente por um usuário
  - Geralmente programas de terceiros
  - Ex.: Desktops, notebooks, tablets, smartphones, etc
  - Compromisso entre custo e desempenho para o usuário
- Embarcados

Principais Classes de Sistemas Computacionais

- Servidores
- Pessoais
- Embarcados
  - Recursos projetados para fins específicos
  - Software de difícil customização, geralmente integrado ao hardware.
  - Ex.: Eletroeletrônicos (TV, DVD, SetupBox, maquina de lavar,...), Automóveis/Barcos/Aviões, Industriais, Brinquedos.
  - Geralmente baixo custo e baixa dependabilidade, embora alguns precisem de baixa taxas de falhas (sistemas redundantes).

Computador

- Componentes básicos de um computador:



Baseado na arquitetura de Von Neumann!

Hardware

- Processador (CPU):
  - É o cérebro do computador, a parte que interpreta e executa instruções. (Um programa = instruções ordenadas logicamente.)
  - O termo CPU (*Central Processing Unit* – Unidade Central de Processamento) é usado genericamente para se referir ao processador de um computador.
  - A CPU não é o gabinete do computador, mas sim um **chip**, que se localiza na placa mãe (**motherboard**) que está dentro do gabinete.

  - Nos computadores pessoais (PC), o processador é composto de um ou mais núcleos de processamento (*cores*)

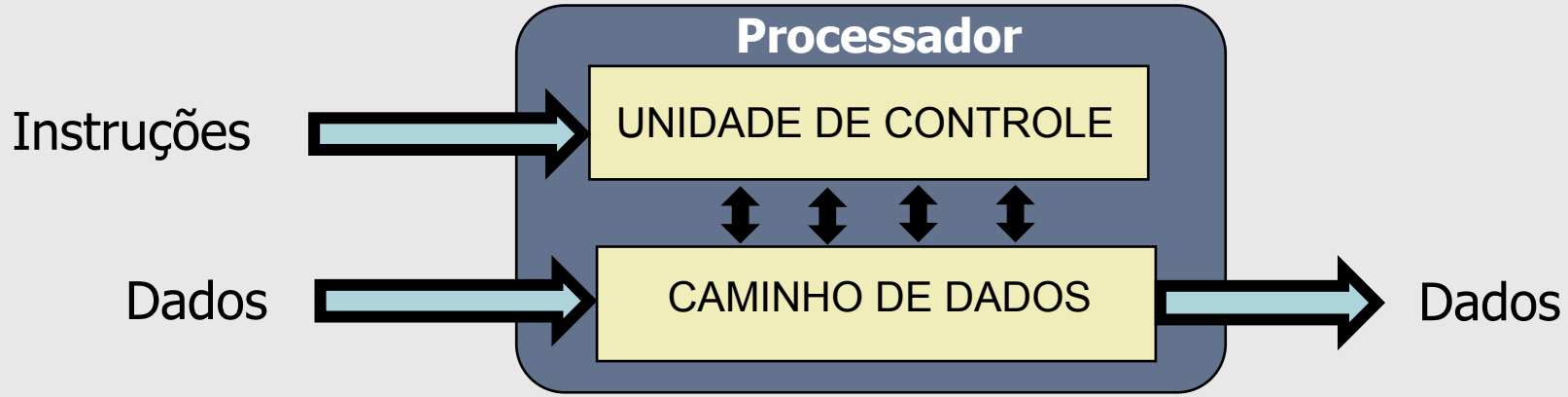
Componentes do Computador

- **Processador**: É dividido em 2 partes principais
  - **UNIDADE DE CONTROLE (*Control*):**

    Decodifica (interpreta) a instrução e gera os sinais de controle para o caminho de dados.

    "É quem manda as coisas acontecerem"
  - **UNIDADE OPERATIVA ou CAMINHO DE DADOS (*Datapath*)**: É controlado pela unidade de controle. Realiza efetivamente o processamento dos dados (operações lógicas, aritméticas, manipulação, acesso à memória e aos dispositivos de E/S).

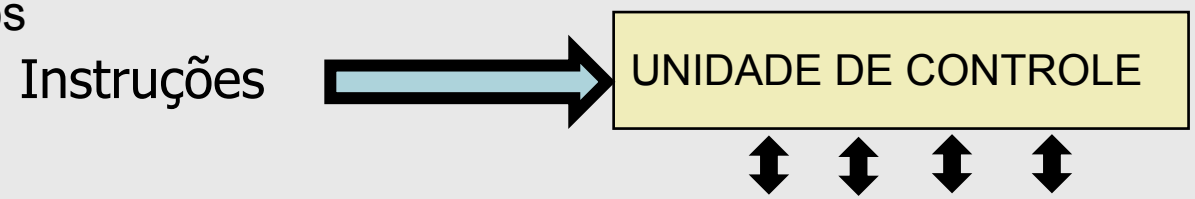
    "É quem faz as coisas acontecerem"



Componentes do Computador

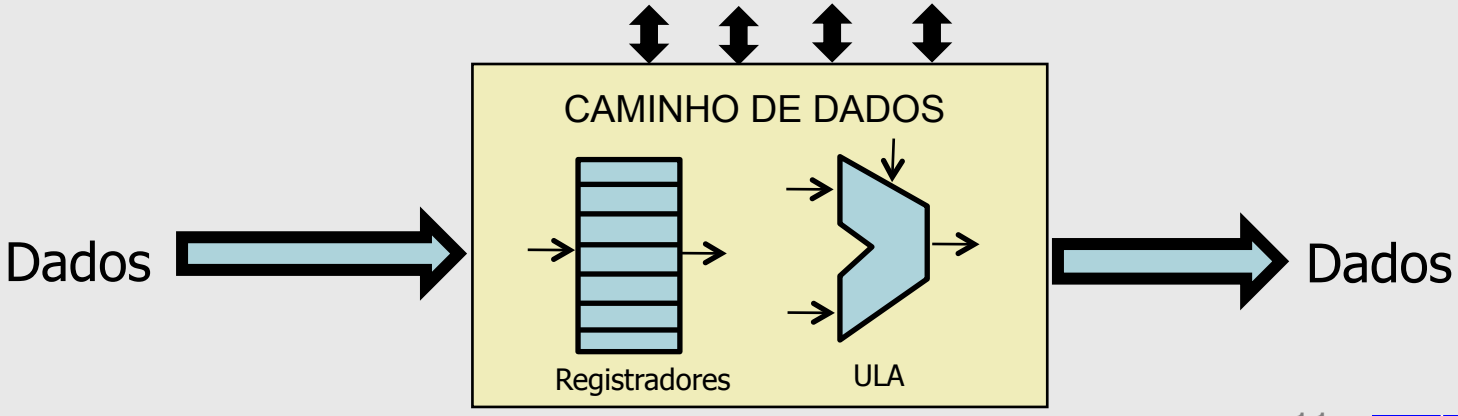
- **UNIDADE DE CONTROLE (*Control*):**

  Composta por circuitos digitais que recebem os bits correspondentes à instrução a ser executada e geram os sinais digitais que comandam o caminho de dados



- **UNIDADE OPERATIVA ou CAMINHO DE DADOS (*Datapath*):**

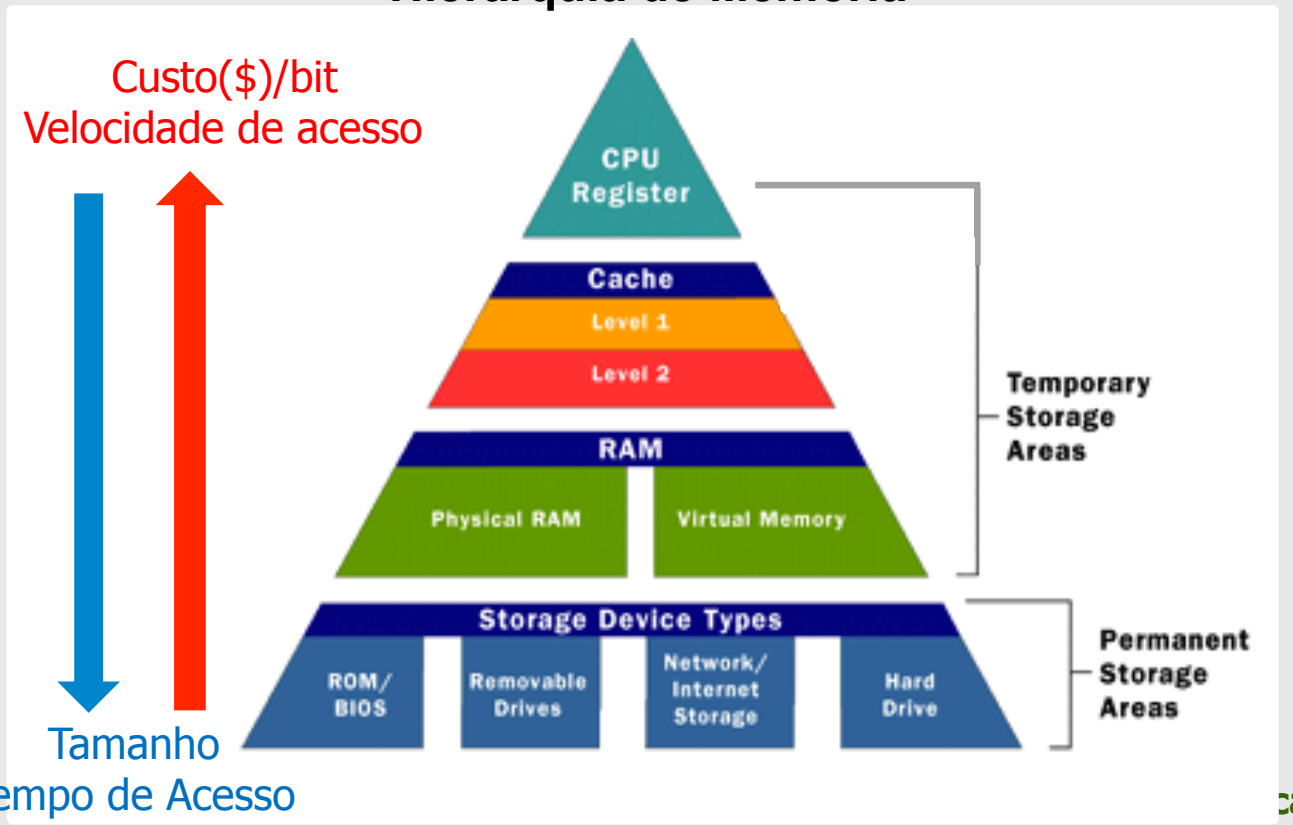
  Composto por Registradores (componente capaz de armazenar um número), Unidade Lógico/Aritmética-ULA (circuito capaz de efetuar operações matemáticas), e dispositivos de acesso à memória e E/S.
  Recebe os sinais digitais de controle e executa a instrução.

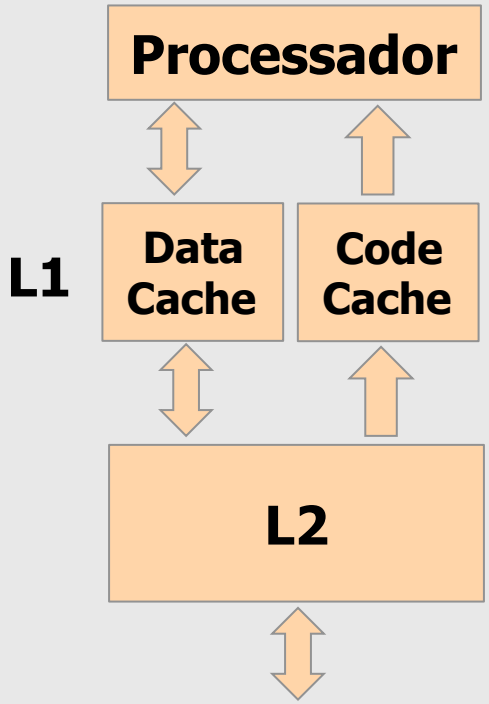


Componentes do Computador

- Dispositivos de Memória
  - Local onde os dados e programas são armazenados.
  - Sem uma memória de onde os processadores podem ler e escrever informações, não haveria nenhum computador digital de programa armazenado.

Hierarquia de Memória





carlacastanho@unb.br

Hardware

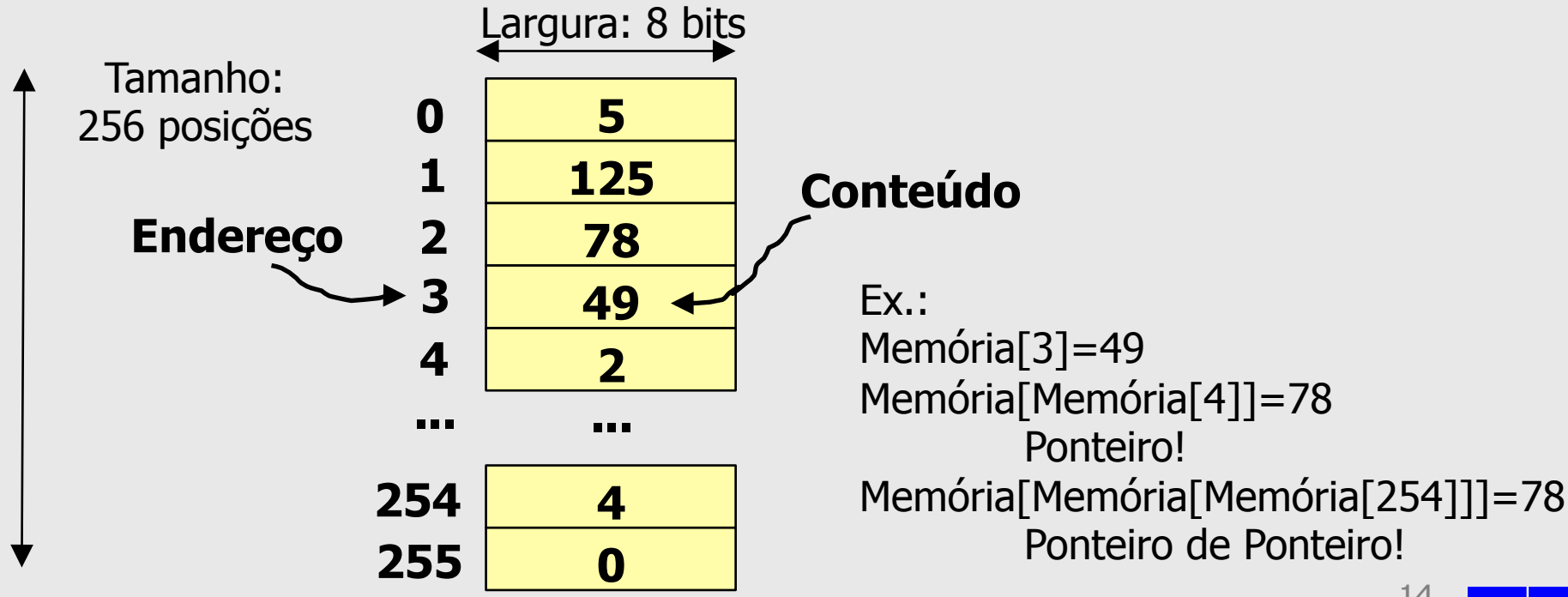
- Dispositivos de Memória: *Obs.: A memória cache é transparente ao programador!*

MEMÓRIA PRINCIPAL/ PRIMÁRIA (RAM – Random Access Memory)	MEMÓRIA AUXILIAR/ SECUNDÁRIA (HDD – Hard Disk Drive)
- Acesso mais rápido, - Capacidade mais restrita. - Armazena informações temporariamente durante um processamento realizado pela CPU. - Volátil	- Acesso mais lento - Capacidade bem maior. - Armazena grande conjunto de dados que a memória principal não suporta. - Não volátil

Os dados e programas devem primeiro ser transferidos da memória auxiliar para a memória principal antes de serem processados

Componentes do Computador

- Memória Principal (RAM)
  - Consiste de um arranjo de diversos elementos capazes de armazenar um determinado número de bits (dado), onde a cada elemento é associado a um endereço (número).
  - Organização mais comum são memórias onde cada elemento possui 8 bits = 1 Byte, mas outras estruturas podem ser usadas.
  - Representação: Ex.: Memória 256 x 8



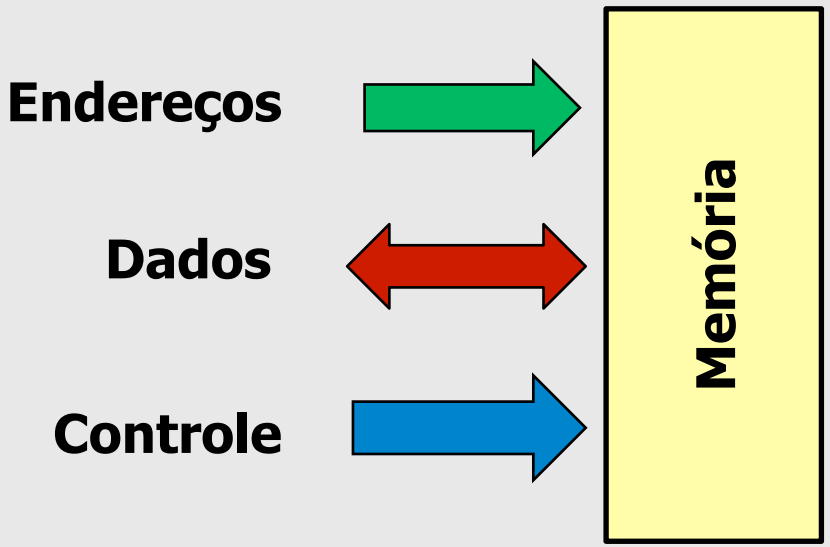
Ex.:
Memória[3]=49
Memória[Memória[4]]=78
Ponteiro!
Memória[Memória[Memória[254]]]=78
Ponteiro de Ponteiro!

APC – carlacastanho@unb.br

Componentes do Computador

As memórias são geralmente acessadas por 3 conjuntos de vias ou barramentos:

- **Barramento de Endereços**
  - **Na leitura:** Contém o endereço de onde o processador vai ler o dado, ou, instrução.
  - **Na escrita:** Contém o endereço onde o processador vai escrever (gravar) o dado.
- **Barramento de Dados**
  - **Na leitura:** Conterá o dado, ou instrução, lido da memória
  - **Na escrita:** Conterá o dado a ser escrito na memória
- **Barramento de Controle**
  - **Na leitura:** Deve configurar a memória para a realização de uma leitura
  - **Na escrita:** Deve configurar a memória para a realização de uma escrita



Hardware

- Dispositivos de Entrada/Saída: (Periféricos)
  - Muitas vezes chamados de dispositivos de I/O (Input/Output)
  - Compreende todas as maneiras como o computador se comunica com os usuários, outras máquinas ou dispositivos.
  - Exemplos:
    - ENTRADA: mouse, teclado, ... (o que mais?)
    - SAÍDA: vídeo, impressora, ... (o que mais?)




Componentes do Computador

- Dispositivos de Comunicação
  - São dispositivos de E/S dedicados a realizar a comunicação entre o computador e outras máquinas/dispositivos/computadores.
  - Exemplos:
    - Modem
    - Interface de Rede Ethernet, Gigabit Ethernet, Fibra ótica,...
    - Interface de Rede Wi-Fi (IEEE 802.11), Wimax,...
    - Interface GPRS (2G), EDGE(2,5G), HSDPA (3G), ...
    - Interface Bluetooth, Zigbee, ...
    - Interface USB (*Universal Serial Bus*), IEEE1394 (*Firewire*), ...
    - Interfaces Serial, Paralela, SCSI, ...
    - Interfaces industriais
    - Etc.

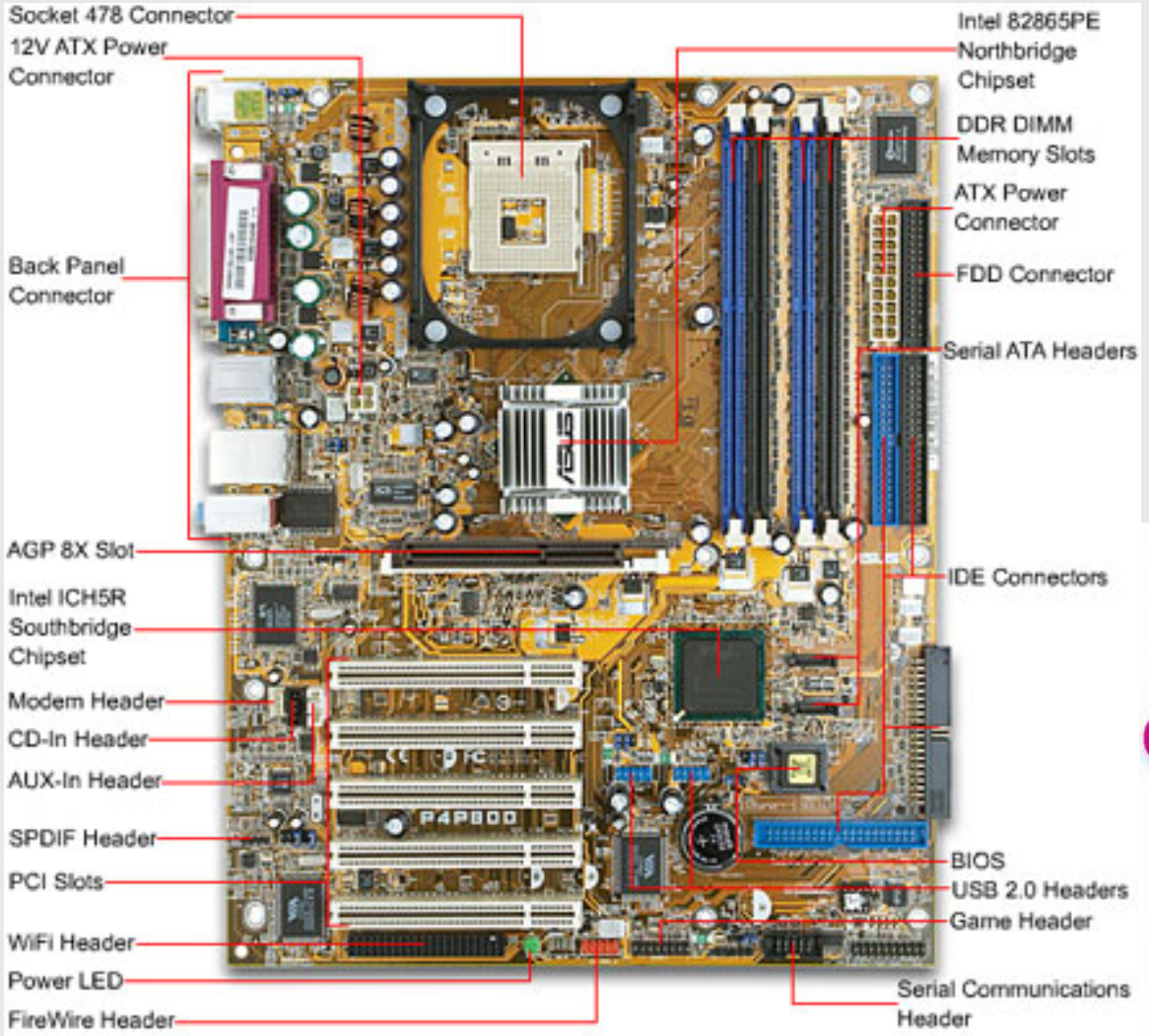
Componentes do Computador

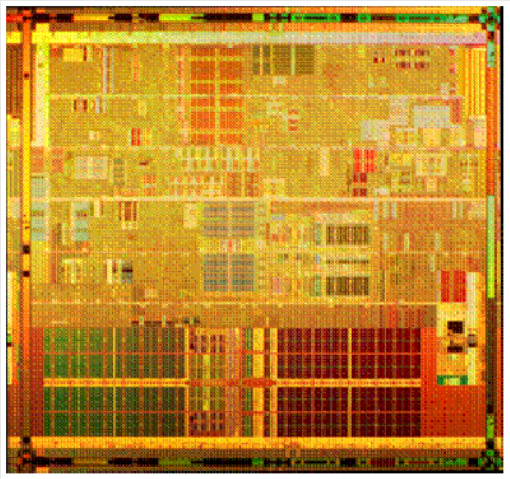
- Placa Mãe (*Mother Board*)
  - É onde se localiza fisicamente o Processador e por onde se conectam todos os dispositivos externos a ele.
    - Dispositivos de Memória
    - Dispositivos de E/S
    - Dispositivos de Comunicação

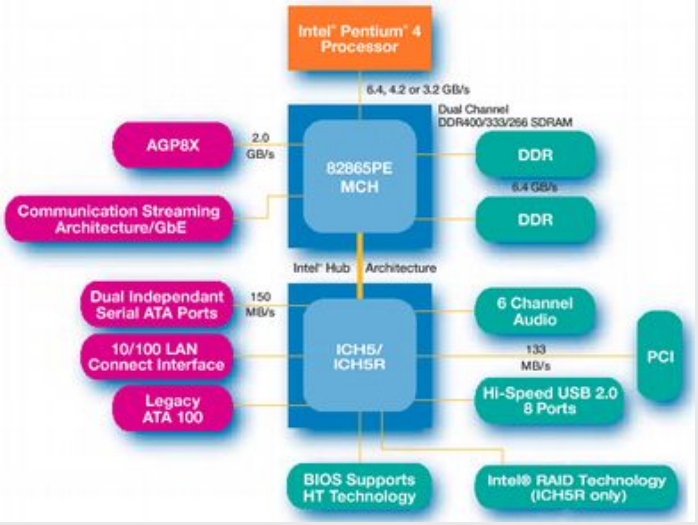
- Fonte de Alimentação
  - Responsável por fornecer a energia necessária para que o hardware funcione, executando o software.
  - Pode ser :
    - Fonte chaveada (ligada na rede de energia elétrica)
    - Bateria, recarregável ou não
  - Consumo e dissipação de calor sempre foram pontos muito importantes na área da computação.

APC – carlacastanho@unb.br

Placa mãe para Pentium IV

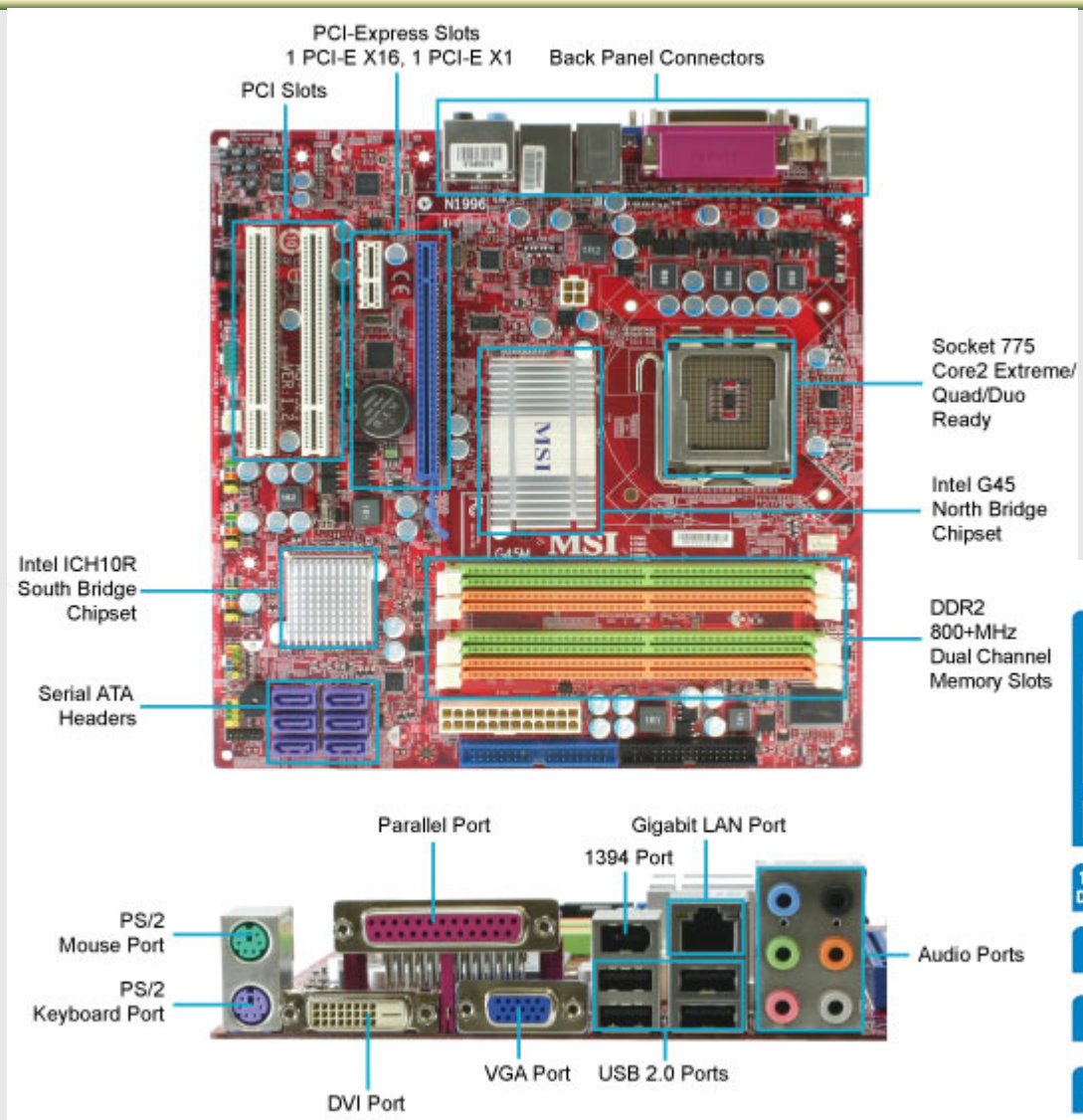


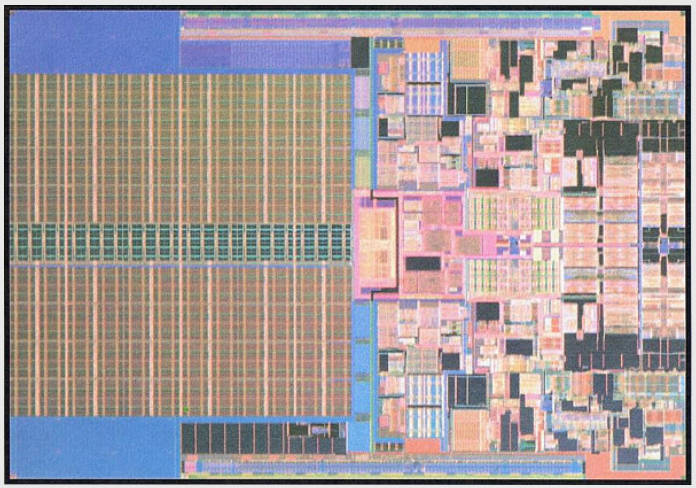


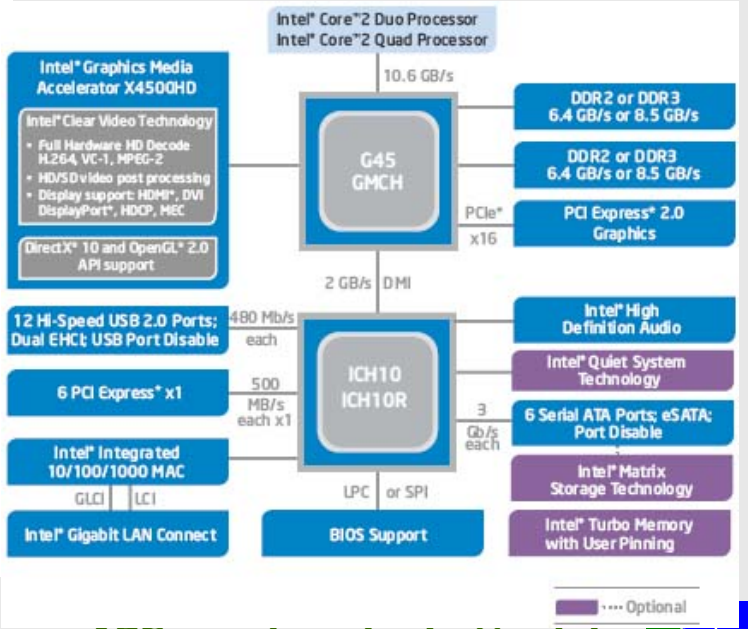


APC – carlacastanho@unb.br

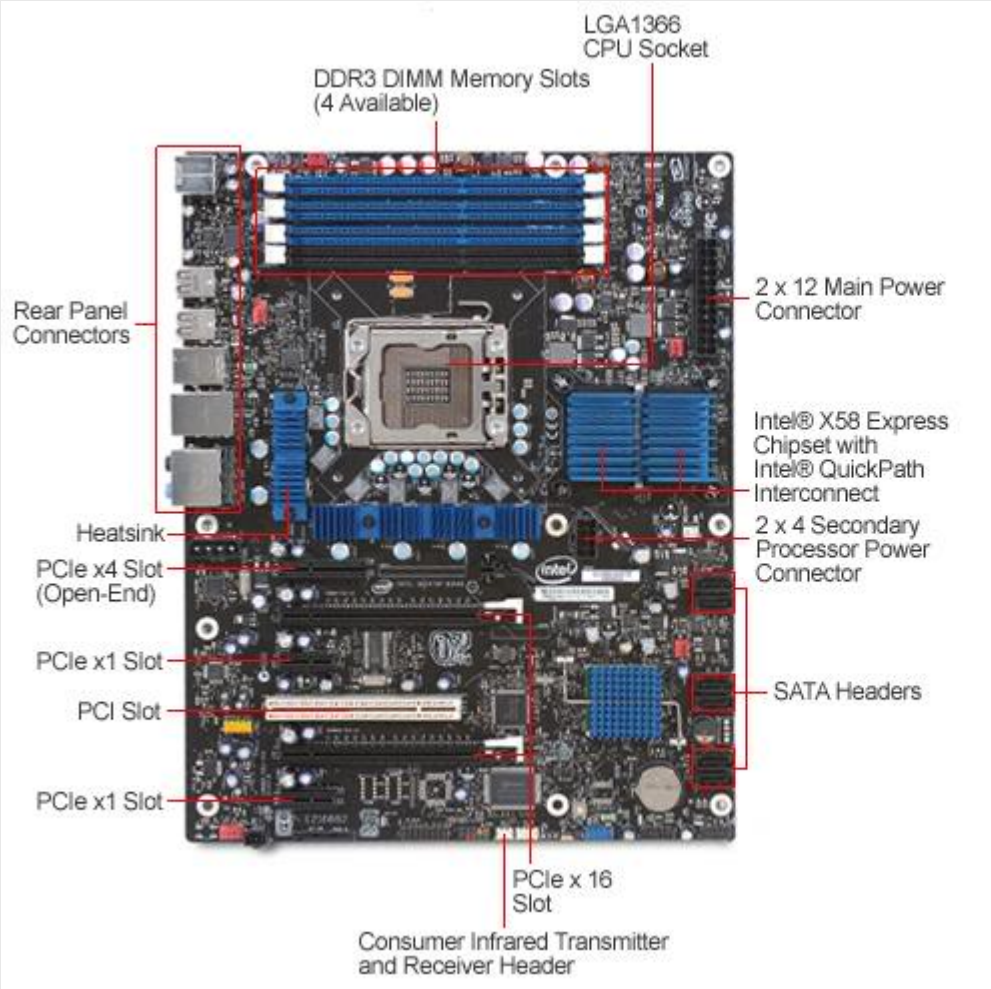
Placa mãe para Core2

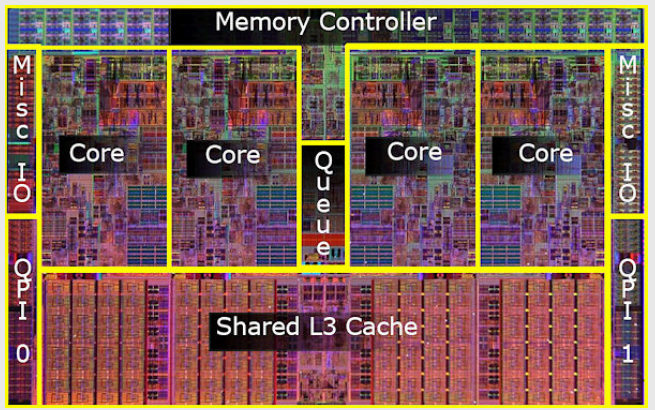


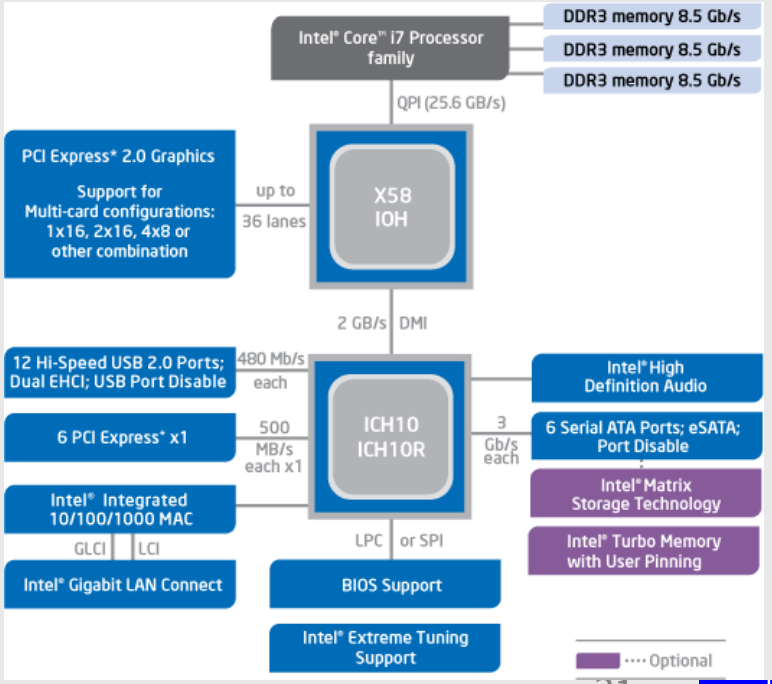




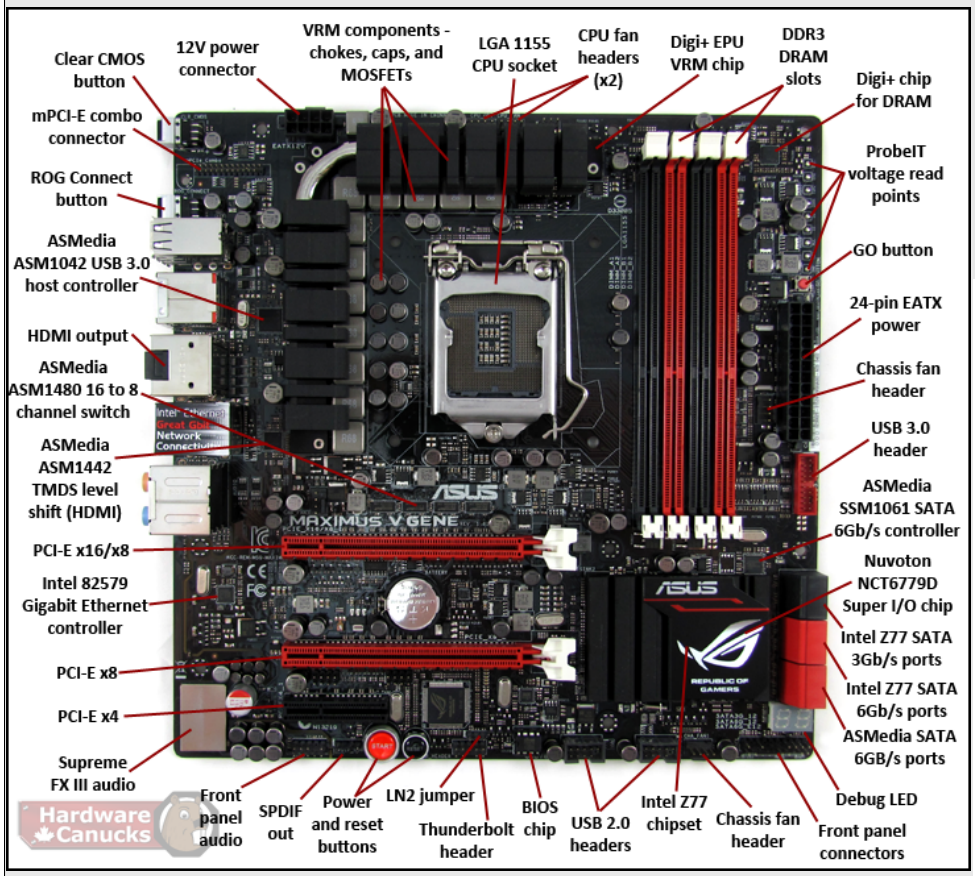
Placa mãe para Core i7

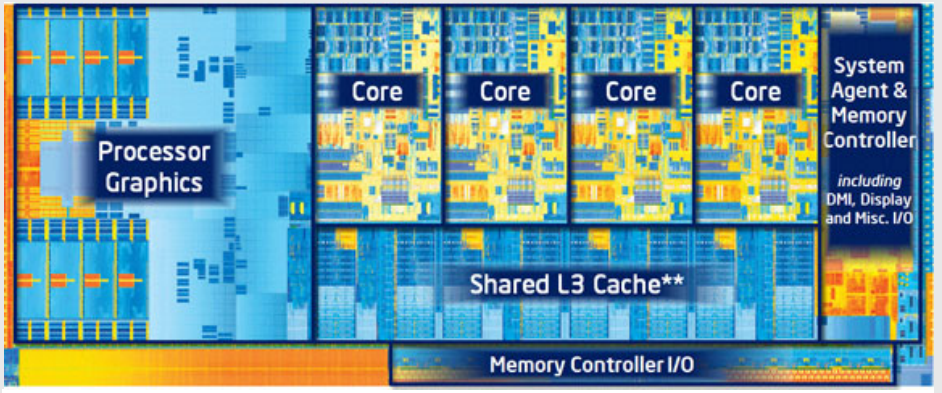


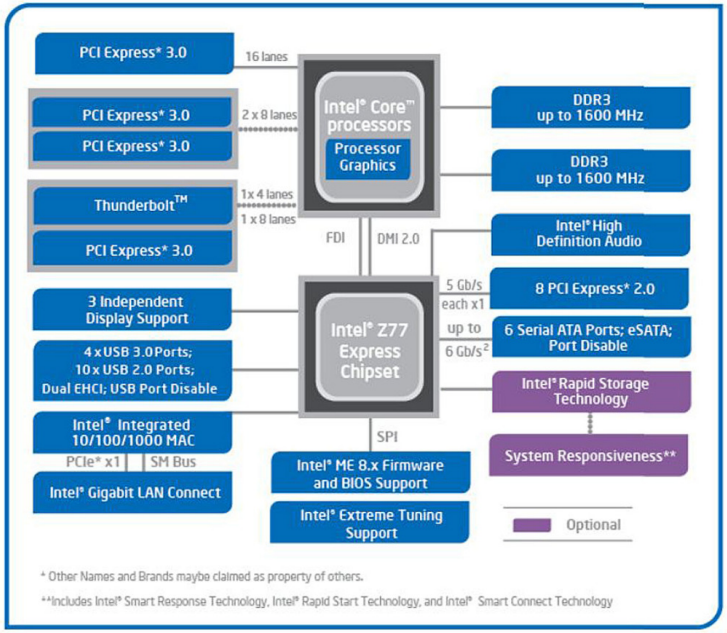




Placa mãe para Core i7 3ª geração







Intel® Z77 Express Chipset Platform Block Diagram

Software

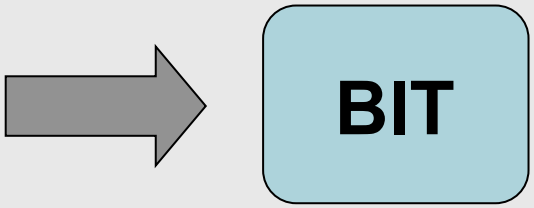
- Software:
  - São instruções escritas em linguagem de programação que dirão ao computador o que fazer e auxiliarão o usuário em suas atividades. Ou seja, os programas e os sistemas de programação utilizados por um computador e que permitem atender às necessidades do usuário.

CLASSIFICAÇÃO
- Sistemas Operacionais
- Compiladores
- Interpretadores
- Utilitários
- Aplicativos
- Gerênciadores de Banco de Dados
- Editores de Texto
- Editores Gráficos
- Planilhas Eletrônicas
- Lazer
- Outros...

A Eletrônica Digital do Computador

- Os circuitos eletrônicos de um computador moderno operam com sinais de dois níveis distintos ou *binário*.
  - Motivo: solução simples e de baixo custo de implementação.
- Ingrediente básico dos CHIPS (pastilhas): *transistor*
  - Transistor: componente básico criado a partir de um material semicondutor, isto é, possui a propriedade de conduzir corrente elétrica após a aplicação de um tensão (chave “liga-desliga”)

Todos os dados armazenados e processados em um computador são traduzidos em sinais elétricos binários, ou seja, em um conjunto finito de 0s e 1s.



Conceitos de bits e seus múltiplos

- *bit* (*binary digit*):
  - representa a forma lógica de um estado “ligado/desligado” ou binário existente em dispositivos eletrônicos digitais dos circuitos de um computador.
  - bit “ligado”é representado pelo símbolo 1.
  - bit “desligado”é representado pelo símbolo 0.

- Em seu nível mais baixo, **tudo** (letras, algarismos, sinais de pontuação, símbolos, comandos) na memória do computador é representado por **números binários**.

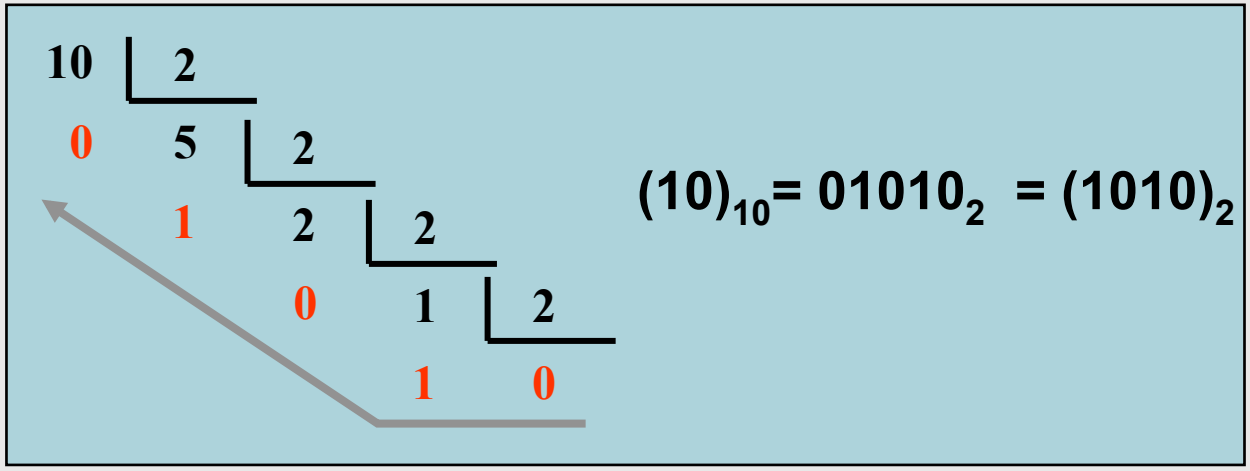
Conceitos de bits e seus múltiplos

- Manipulação de números binários:
  - A posição de cada dígito de um número representa a potência da base 2.

Exemplo de conversão de um número binário em decimal:

$$(10101)_2 = 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = (21)_{10}$$

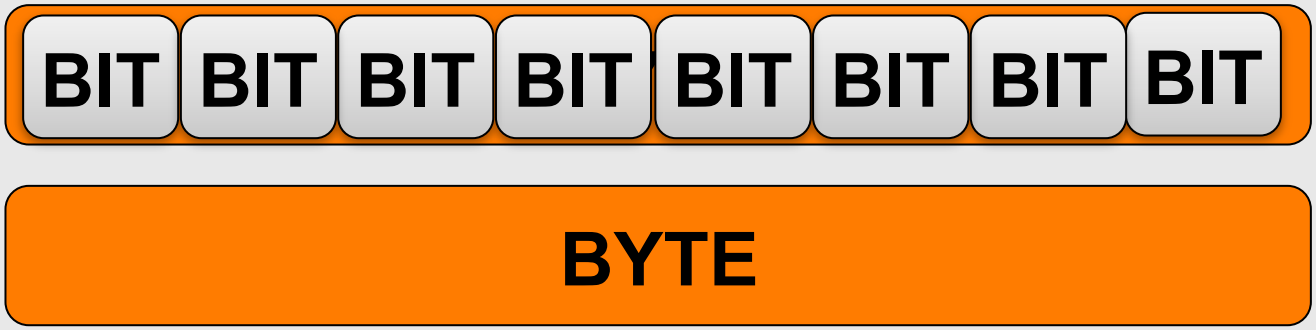
Exemplo de conversão de um número decimal em binário:



$$(10)_{10} = 01010_2 = (1010)_2$$

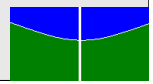
Conceitos de bits e seus múltiplos

- Embora a unidade fundamental de informação do computador seja o bit, na prática utilizamos seus múltiplos, como o **BYTE**:
  - É um conjunto de 8 bits.
  - Para fins de programação o BYTE é o menor dado que se pode manipular diretamente.



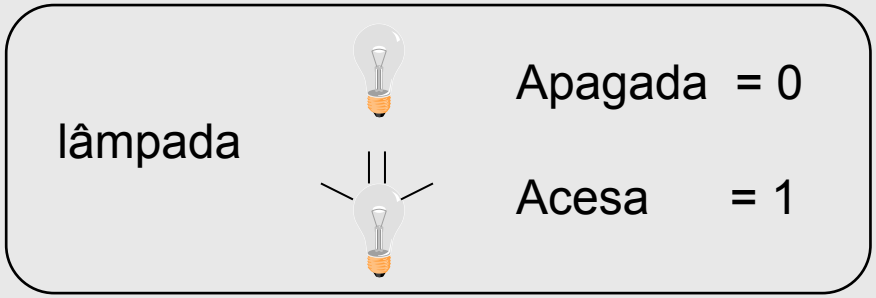
Conceitos de bits e seus múltiplos

Quantos números binários diferentes é possível representar utilizando um conjunto de 8 bits (1 byte) ?



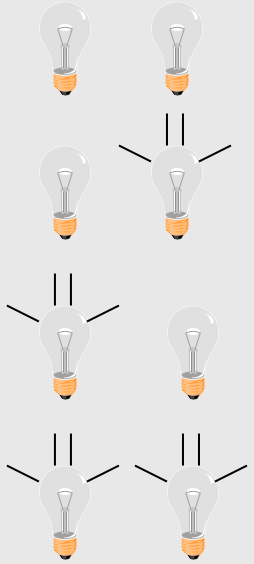
Conceitos de bits e seus múltiplos

- Com 2 bits é possível representar 4 números binários diferentes:



Conjunto de 2 lâmpadas

2^2 = 4 combinações



Portanto, com 1 byte (8 bits) é possível representar 2^8 = 256 números binários diferentes

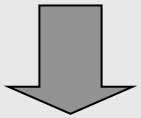
Conceitos de bits e seus múltiplos

- Unidades de medida:
  - Tanto para quantificar a memória principal do equipamento como para medir a capacidade de armazenamento, são usados múltiplos de bytes, como “K”, “M”, “G”, e “T”, respectivamente **Kilo (mil)**, **Mega (milhão)**, **Giga (bilhão)**, e **Tera (trilhão).**

Os múltiplos do byte			
1 Kilobyte (Kbyte ou KB)	2^{10}	1024 bytes	$\approx 10^3$ bytes
1 Megabyte (Mbyte ou MB)	2^{20}	1.048.576 bytes	$\approx 10^6$ bytes
1 Gigabyte (Gbyte ou GB)	2^{30}	1.073.741.824 bytes	$\approx 10^9$ bytes
1 Terabyte (Tbyte ou TB)	2^{40}	1.099.511.627.776 bytes	$\approx 10^{12}$ bytes

ASCII

- A representação de símbolos no computador, além dos próprios números é conseguida associando-se sequências de bits a cada caracter particular.
- Por necessidade de diálogos entre os diferentes computadores, foi criado um **código** utilizado pela maioria dos fabricantes.



ASCII
(American Standard Code Information Interchange)

ASCII: Define uma tabela de equivalência entre um byte (8 bits) e um símbolo (caracteres alfabéticos, maiúsculos e minúsculos, algarismos, caracteres especiais, símbolos gráficos, de controle do computador, letras gregas e caracteres de acentuação).

ASCII

- Tabela ASCII

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

O conjunto de códigos ASCII original possuia 128 símbolos.

ASCII

- Tabela ASCII Extendida

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
128	80	Ç	160	A0	á	192	C0	└	224	E0	α
129	81	ü	161	A1	í	193	C1	┴	225	E1	ß
130	82	é	162	A2	ó	194	C2	┬	226	E2	Γ
131	83	â	163	A3	ú	195	C3	├	227	E3	π
132	84	ä	164	A4	ñ	196	C4	─	228	E4	Σ
133	85	à	165	A5	Ñ	197	C5	┼	229	E5	σ
134	86	å	166	A6	ª	198	C6	╞	230	E6	µ
135	87	ç	167	A7	º	199	C7	╟	231	E7	τ
136	88	ê	168	A8	¿	200	C8	╚	232	E8	Φ
137	89	ë	169	A9	⌐	201	C9	╔	233	E9	Θ
138	8A	è	170	AA	¬	202	CA	╩	234	EA	Ω
139	8B	ï	171	AB	½	203	CB	╦	235	EB	δ
140	8C	î	172	AC	¼	204	CC	╠	236	EC	∞
141	8D	ì	173	AD	¡	205	CD	═	237	ED	ø
142	8E	Ä	174	AE	«	206	CE	╬	238	EE	ε
143	8F	Å	175	AF	»	207	CF	╧	239	EF	∩
144	90	É	176	B0	░	208	D0	╨	240	F0	≡
145	91	æ	177	B1	▒	209	D1	╤	241	F1	±
146	92	Æ	178	B2	▓	210	D2	╥	242	F2	≥
147	93	ô	179	B3	│	211	D3	╙	243	F3	≤
148	94	ö	180	B4	┤	212	D4	╘	244	F4	⌠
149	95	ò	181	B5	╡	213	D5	╒	245	F5	⌡
150	96	û	182	B6	╢	214	D6	╓	246	F6	÷
151	97	ù	183	B7	╖	215	D7	╫	247	F7	≈
152	98	ÿ	184	B8	╕	216	D8	╪	248	F8	°
153	99	Ö	185	B9	╣	217	D9	┘	249	F9	∙
154	9A	Ü	186	BA	║	218	DA	┌	250	FA	·
155	9B	¢	187	BB	╗	219	DB	█	251	FB	√
156	9C	£	188	BC	╝	220	DC	▄	252	FC	ⁿ
157	9D	¥	189	BD	╜	221	DD	▌	253	FD	²
158	9E	₧	190	BE	╛	222	DE	▐	254	FE	■
159	9F	ƒ	191	BF	┐	223	DF	▀	255	FF	

O conjunto de códigos ASCII atual possui 256 símbolos.

UNICODE

- UNICODE:
  - É o padrão universal de codificação de caracteres
  - O Unicode fornece um número único para cada caractere, não importando a plataforma (a máquina e/ou sistema operacional em uso), o programa ou o idioma.
  - Foi desenvolvido para resolver problemas que existiam com outros sistemas de codificação, pois não eram suficientes para suportar todos os caracteres e idiomas existentes.
  - Sua criação foi baseada na tabela ASCII.
  - Permite definir caracteres cuja representação interna no computador utiliza mais de um byte - 16 bits (UTF-16) e 32 bits (UTF-32).
  - Vários sistemas operacionais, programas e browsers modernos suportam o Unicode.

Sistemas de Numeração

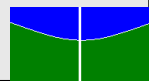
- Quando nós, seres humanos, trabalhamos com números, utilizamos a base 10, também chamada de decimal.

- **DECIMAL**:
  - Base: 10 (quantidade de símbolos).
  - Elementos: 0, 1, 2, 3, 4, 5, 6, 7, 8 e 9.
  - Embora o Sistema Decimal possua somente dez símbolos, qualquer número acima disso pode ser expresso usando o sistema de peso por posicionamento, conforme o exemplo a seguir:
    - $(1967)_{10} = 1x10^3 + 9x10^2 + 6x10^1 + 7x10^0$

Sistemas de Numeração

- O computador trabalha com outro sistema, o Binário, pois se um computador trabalhasse com a base dez, seus circuitos seriam ainda mais complicados.

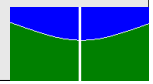
- **BINÁRIO:**
  - Base: 2 (quantidade de símbolos).
  - Elementos: 0, 1
  - Exemplos: $(11011)_2$ $(1011)_2$ $(100101000)_2$



Sistemas de Numeração

- Com o propósito de minimizar a representação de um número binário e facilitar a manipulação humana, foi criado o sistema Hexadecimal.

- **HEXADECIMAL:**
  - Base: 16 (quantidade de símbolos).
  - Elementos: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E e F.
  - Exemplos: $(23)_{16}$ $(1A3F)_{16}$ $(12BD3F4)_{16}$



Sistemas de Numeração

- **HEXADECIMAL:**
  - Se considerarmos quatro dígitos binários, ou seja, quatro bits, o maior número que se pode expressar com esses quatro dígitos é 1111, que é, em decimal 15.
  - Como não existem símbolos dentro do sistema arábico, que possam representar os números decimais entre 10 e 15, sem repetir os símbolos anteriores, foram usados símbolos literais: A, B, C, D, E e F.
  - Dois dígitos hexadecimais representam os números de 0 a 255 (em binário, 8 bits).

Base-10	Base-2	Base-16
Decimal	Binário	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F
16	0001 0000	10
17	0001 0001	11
18	0001 0010	12
31	0001 1111	1F
100	0110 0100	64
255	1111 1111	FF

Sistemas de Numeração

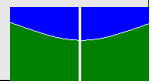
- Em projetos de informática (isto é, nos trabalhos realizados pelos programadores, analistas e engenheiros de sistemas), é usual utilizar o sistema hexadecimal para reduzir o número de algarismos da representação e consequentemente facilitar a compreensão da grandeza e evitar erros.
  - Exemplo:
    - Em decimal:

      2.780.898.547
    - Em Binário:

      10100101111000001001001011110011

      1010 0101 1100 0001 0010 0100 1111 0011
    - Em hexadecimal:

      A5 C1 24 F3.



Sistemas de Numeração

- Na linguagem HTML, por exemplo, as cores são especificadas em hexadecimal.
  - O código RGB (**R**ed - **G**reen - **B**lue) informa a quantidade de luz vermelha, verde e azul que compõe a cor, respectivamente.
  - Este valor é representado em número hexadecimal, onde os bytes que variam de 00 (ausência da cor) a FF (maior intensidade da cor), estão divididos em três grupos. Cada grupo pode variar até 256 tons da cor que ele representa. Os tons podem ser misturados com os tons de outras cores e o total de combinações possíveis é de **256 x 256 x 256 = 16.777.216**.
  - Exemplo:

    `#FF0000` **é vermelho**
    `#00FF00` **é verde**
    `#0000FF` **é azul**
    `#000000` **é preto (ausência das cores)**
    `#FFFFFF` **é branco(a soma de todas elas)**

Sistemas de Numeração

- Conversão entre os sistemas de numeração
  - Base 2 => Base 10

$$(11011)_2 = (1x2^4 + 1x2^3 + 0x2^2 + 1x2^1 + 1x2^0)_{10}$$
$$= (16 + 8 + 0 + 2 + 1)_{10}$$
$$= (27)_{10}$$

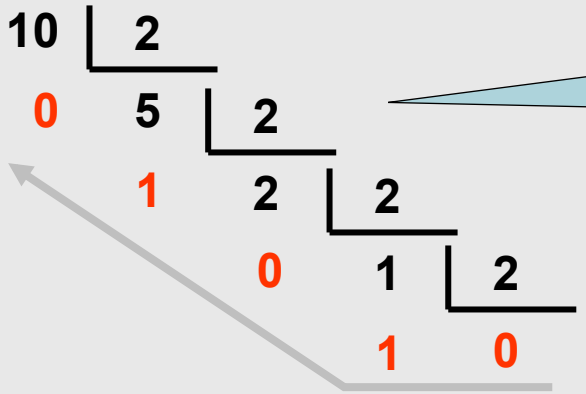
$$(111)_2 = (1x2^2 + 1x2^1 + 1x2^0)_{10}$$
$$= (4 + 2 + 1)_{10}$$
$$= (7)_{10}$$

$$(111,01)_2 = (1x2^2 + 1x2^1 + 1x2^0 + 0x2^{-1} + 1x2^{-2})_{10}$$
$$= (4 + 2 + 1 + 0 + 0,25)_{10}$$
$$= (7,25)_{10}$$

Quando houver parte fracionária, os expoentes vão crescendo negativamente.

Sistemas de Numeração

- Conversão entre os sistemas de numeração
  - Base 10 => Base 2



Dividir sucessivamente por 2 o número decimal e os quocientes que vão sendo obtidos, até que o quociente de uma das divisões seja 0.

$(10)_{10} = 01010 = (1010)_2$

$(5,25)_{10}$ = 101 (parte inteira na base 2)

Parte fracionária:

0,25 x 2 = 0,50 => 0

0,50 x 2 = 1,00 => 1

$(5,25)_{10} = (101,01)_2$

Multiplica da parte fracionária pela base de destino tantas vezes quantas casas decimais se desejar; a cada multiplicação, pega-se o dígito que passa para a esquerda da vírgula, volta-se a pegar apenas as casas decimais restantes e prossegue-se até zerar o resultado ou até atingir a aproximação desejada.

Sistemas de Numeração

- Conversão entre os sistemas de numeração
  - Base 16 => Base 10

Quando houver parte fracionária, a conversão é feita da mesma forma apresentada anteriormente, apenas alterando a base.

$$(17)_{16} = (1x16^1 + 7x16^0)_{10} = (16 + 7)_{10} = (23)_{10}$$

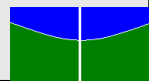
$$(C203)_{16} = (12x16^3 + 2x16^2 + 0x16^1 + 3x16^0)_{10} = (49152 + 512 + 0 + 3)_{10} = (49667)_{10}$$

  - Base 10 => Base 16

Dividir sucessivamente por 16 o número decimal e os quocientes que vão sendo obtidos, até que o quociente de uma das divisões seja 0.

49667	16						
3	3104	16					
	0	194	16				
		2	12	16			
			12	0			

$(49667)_{10} = (C203)_{16}$



Sistemas de Numeração

- Conversão entre os sistemas de numeração
  - Base 2 => Base 16 (Base 16 => Base 2)
    - Separamos o número binário em grupos de 4 dígitos e substituímos cada grupo pelo dígito hexadecimal correspondente (cada dígito hexadecimal pelo número binário correspondente).

$$(1000 \quad 0111 \quad 0100 \quad 0010)_2$$
$$(\quad 8 \quad\quad 7 \quad\quad 4 \quad\quad 2 \quad)_{16}$$

$$(\quad 9 \quad\quad D \quad\quad 8 \quad\quad F \quad)_{16}$$
$$(1001 \quad 1101 \quad 1000 \quad 1111)_2$$

Sistema Operacional

- SISTEMA OPERACIONAL:
  - O sistema operacional cria um ambiente onde os usuários podem preparar seus programas e executá-los sem se preocupar com detalhes de hardware.
  - Um conjunto de programas, que desempenham rotinas necessárias ao funcionamento do computador, tais como:
    - gerenciamento da memória
    - administração dos dados
    - acionamento dos dispositivos
    - execução de programas utilitários

Exemplos:
Linux, Unix, Windows, DOS, MAC OS X

  - Pode ser considerado um intérprete e um gerenciador das atividades realizadas entre o usuário e o computador/hardware.
  - Cada SO é desenvolvido em consonância com as características de determinado microprocessador.

Linguagens de Programação

- LINGUAGEM DE PROGRAMAÇÃO:
  - É um conjunto de termos (vocabulário) e regras (sintaxe) que permitem a formulação de instruções a um computador.
  - Permite construir programas para a resolução de problemas, (construção de aplicativos, utilitários e até de sistemas operacionais).
  - Existem várias linguagens diferentes, cada uma com recursos que facilitam aplicações específicas.

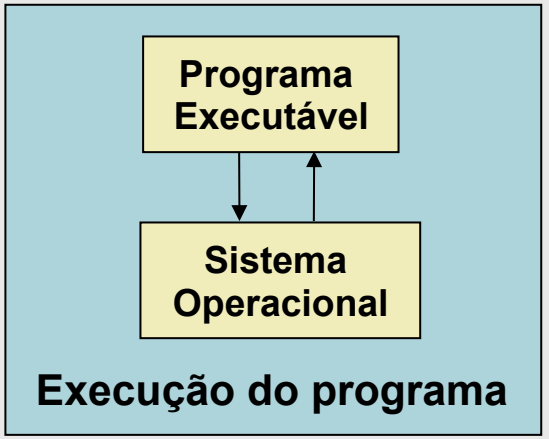
Para um programador é mais importante compreender
os fundamentos e técnicas da programação
do que dominar esta ou aquela linguagem.

Linguagens de Programação

Linguagem de Máquina	Linguagem de Baixo Nível	Linguagem de Alto Nível
- Um programa escrito em linguagem de máquina consiste de uma série de números binários e é muito difícil de ser entendido pelas pessoas. - Uma CPU somente compreende instruções na sua linguagem de máquina.	- São linguagens de programação nas quais os programas são escritos em uma notação que está próxima da linguagem de máquina - Instruções fornecidas pelo fabricante, diferentes para cada computador.	- São linguagens de programação nas quais se pode escrever programas em uma notação próxima à maneira natural de expressar o problema que se deseja resolver.
	Ex: Assembly	Ex: Delphi, Visual Basic, Pascal, C, C++, Java, etc.

Compiladores e Interpretadores

- COMPILADOR
  - Programa utilizado pelo computador para traduzir os comandos simbólicos de uma linguagem de alto nível, para linguagem de máquina (código executável).

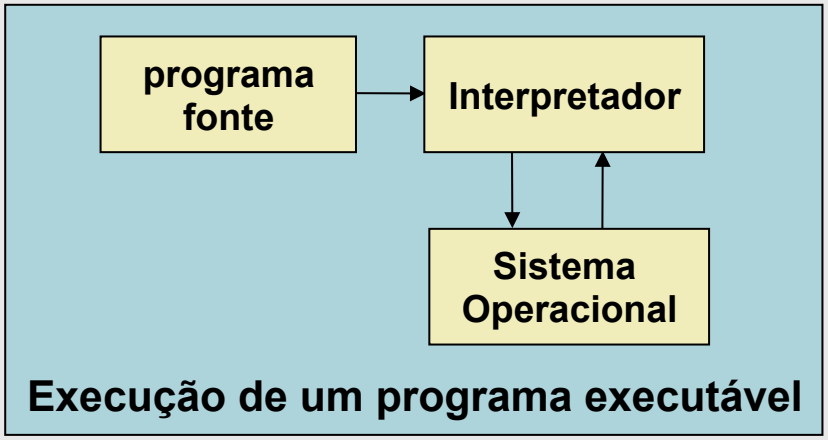


Aparecem nesse processo dois tipos de ERROS, cuja correção consiste em boa parte da tarefa do programador:
- **Erros de compilação**: sintaxe errada, que são mais fáceis de corrigir;
- **Erros de execução**: podem ser fáceis como uma divisão por zero, ou podem ser mais difíceis de corrigir, originados por erros de raciocínio na elaboração do programa.

Exemplos de linguagens Compiladas:
C e Pascal

Compiladores e Interpretadores

- INTERPRETADOR
  - Lê e executa uma declaração do programa por vez. Nenhuma fase intermediária de compilação é necessária. A execução do programa interpretado requer que o interpretador da linguagem esteja sendo executado no computador.



Execução de um programa executável

Exemplos de linguagens interpretadas:
Javascript, Python, Perl