

O COMPUTADOR

Introdução à Computação

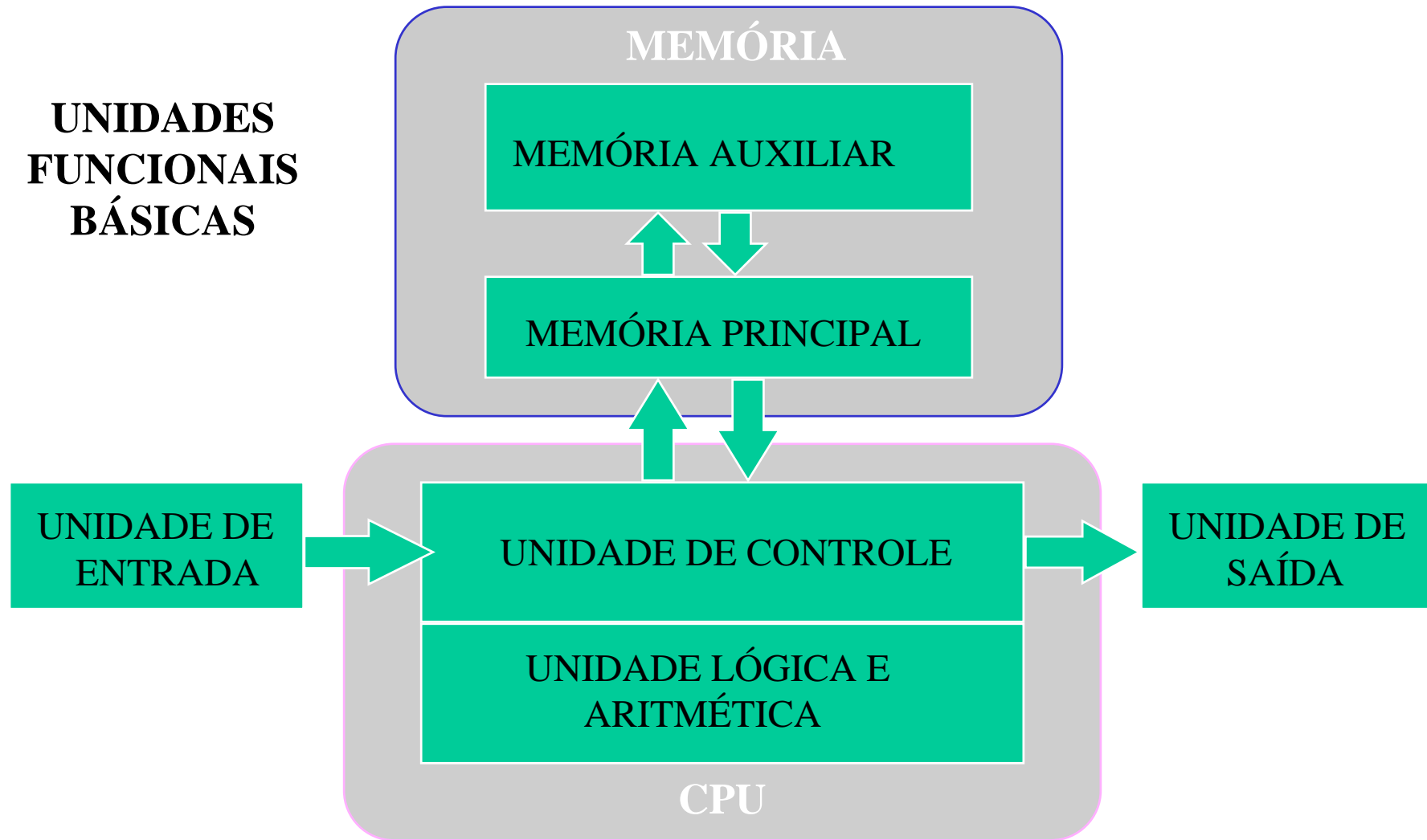
Sumário

- O Hardware
- O Software
- Linguagens de Programação
- Histórico da Linguagem C

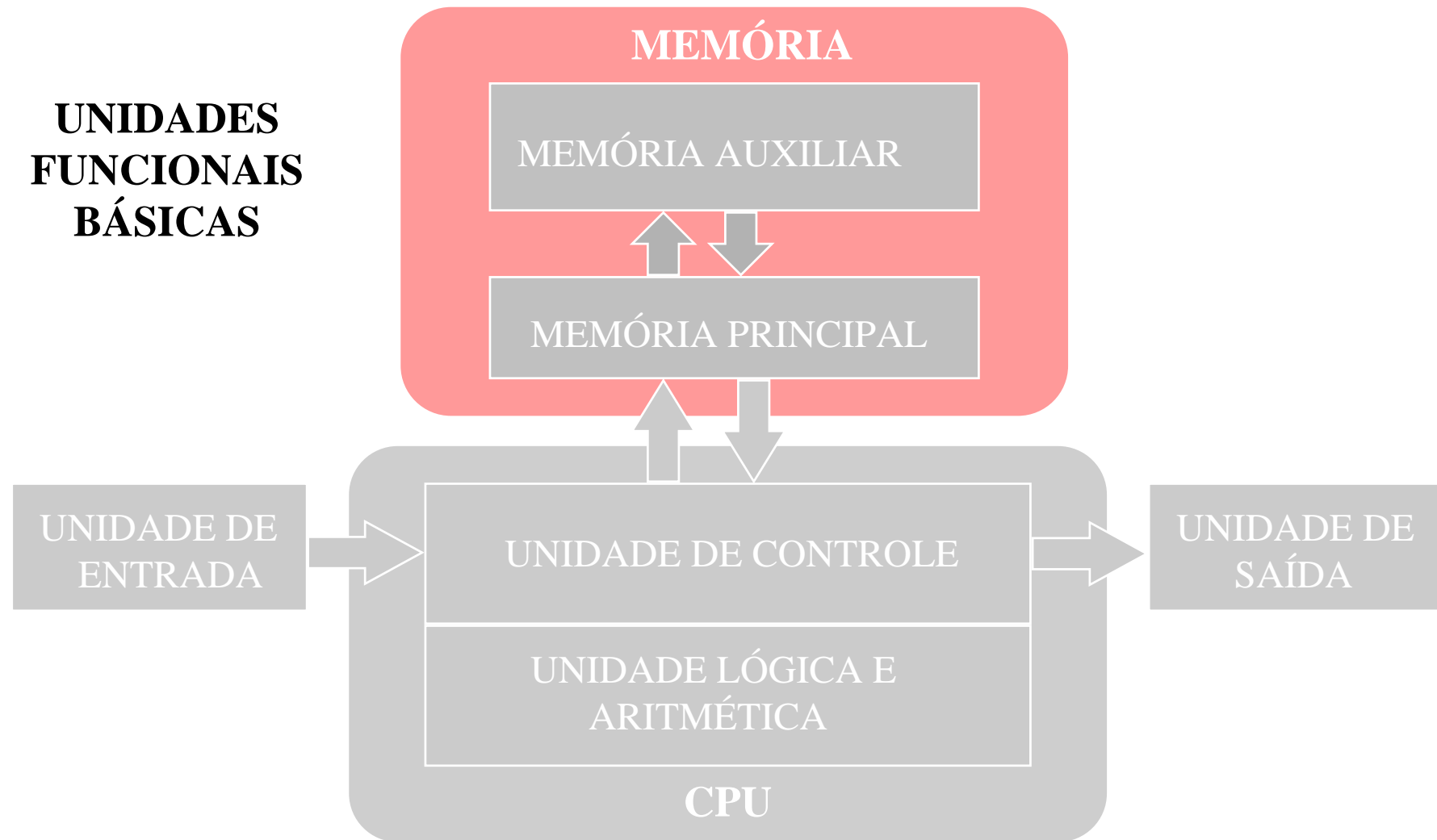
Componentes Básicos do Computador

- O HARDWARE: O equipamento propriamente dito.
Inclui: periféricos de entrada e saída; a máquina e seus elementos físicos: torre, placas, fios, componentes em geral
- O SOFTWARE: Constituído pelos programas (e documentações) que permitem atender às necessidades do usuário

O HARDWARE



A MEMÓRIA DO COMPUTADOR



A MEMÓRIA DO COMPUTADOR

- A memória do computador é dividida em unidades pequenas e de mesmo tamanho, chamadas PALAVRAS, sendo que cada uma tem um único endereço.
- Os endereços são permanentes (vêm da fábrica) e não podem ser modificados pelo programador

	PALAVRA	PALAVRA
endereço	00	04
	PALAVRA	PALAVRA
endereço	08	12

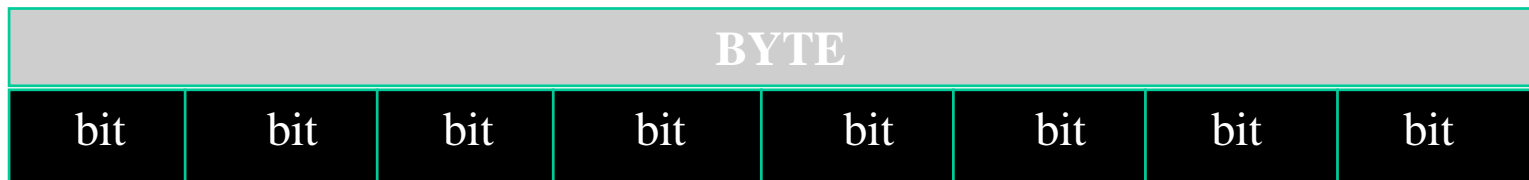
A MEMÓRIA DO COMPUTADOR

- A PALAVRA é formada por um grupo de 2, 4, 6 e até 8 BYTES (depende da arquitetura do computador).
- Exemplo: Palavra de 4 bytes

	PALAVRA				PALAVRA			
	byte	byte	byte	byte	byte	byte	byte	byte
endereço	00	01	02	03	04	05	06	07
endereço	00				04			

A MEMÓRIA DO COMPUTADOR

- BYTE (binary term) : Unidade básica da informação.
- O byte é composto por 8 BITS
- BIT (*binary digit*) - dígitos binários



A MEMÓRIA DO COMPUTADOR

Código ASCII (7 bits)

- Cada byte armazena um caractere: algarismo, letra, símbolo ou caractere de controle.

CARACTERE	ZONA			PARTE NUMÉRICA				
0	0	1	1	0	0	0	0	48
9	0	1	1	1	0	0	1	57
A	1	0	0	0	0	0	1	65
a	1	1	0	0	0	0	1	97
P	1	0	1	0	0	0	0	80
+	0	1	0	1	0	1	1	43
	BYTE							

Tipos de Dados

TABLE I-1 ASCII codes

	0	1	2	3	4	5	6	7	8	9
0	\000	\001	\002	\003	\004	\005	\006	\a	\b	\t
10	\n	\v	\f	\r	\016	\017	\020	\021	\022	\023
20	\024	\025	\026	\027	\030	\031	\032	\033	\034	\035
30	\036	\037	<i>space</i>	!	"	#	\$	%	&	'
40	()	*	+	,	-	.	/	0	1
50	2	3	4	5	6	7	8	9	:	;
60	<	=	>	?	@	A	B	C	D	E
70	F	G	H	I	J	K	L	M	N	O
80	P	Q	R	S	T	U	V	W	X	Y
90	Z	[\]	^	_	`	a	b	c
100	d	e	f	g	h	i	j	k	l	m
110	n	o	p	q	r	s	t	u	v	w
120	x	y	z	{		}	~	\177		

A MEMÓRIA DO COMPUTADOR

Código ASCII (7 bits)

- Possibilidade de 2^7 representações diversas (128 caracteres)
 - alfabeto inglês em letras minúsculas e maiúsculas (52)
 - caracteres decimais numéricos (10)
 - caracteres especiais e de operação (33)
 - caracteres de controle (33)

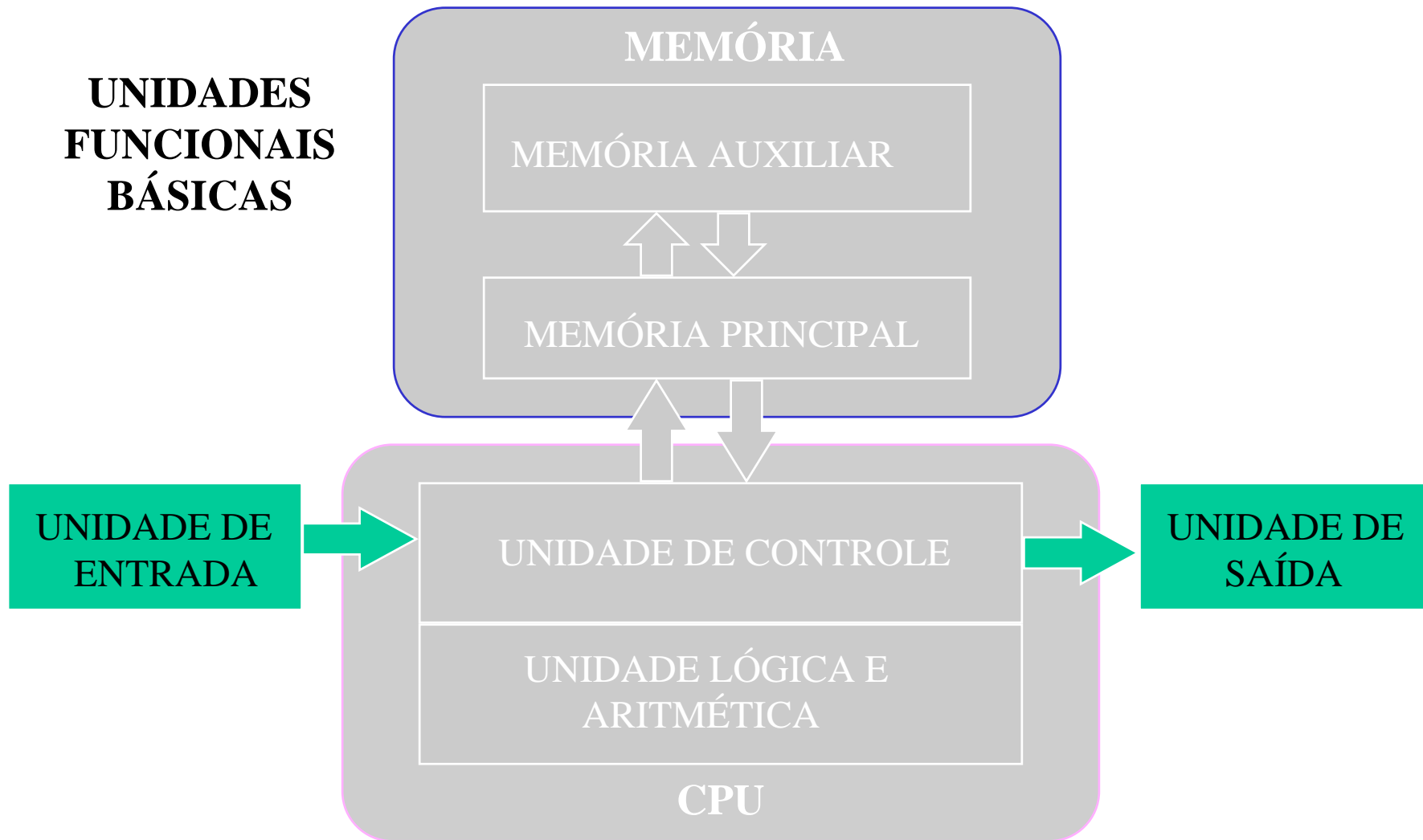
A MEMÓRIA DO COMPUTADOR

UNIDADES DE MEDIDA

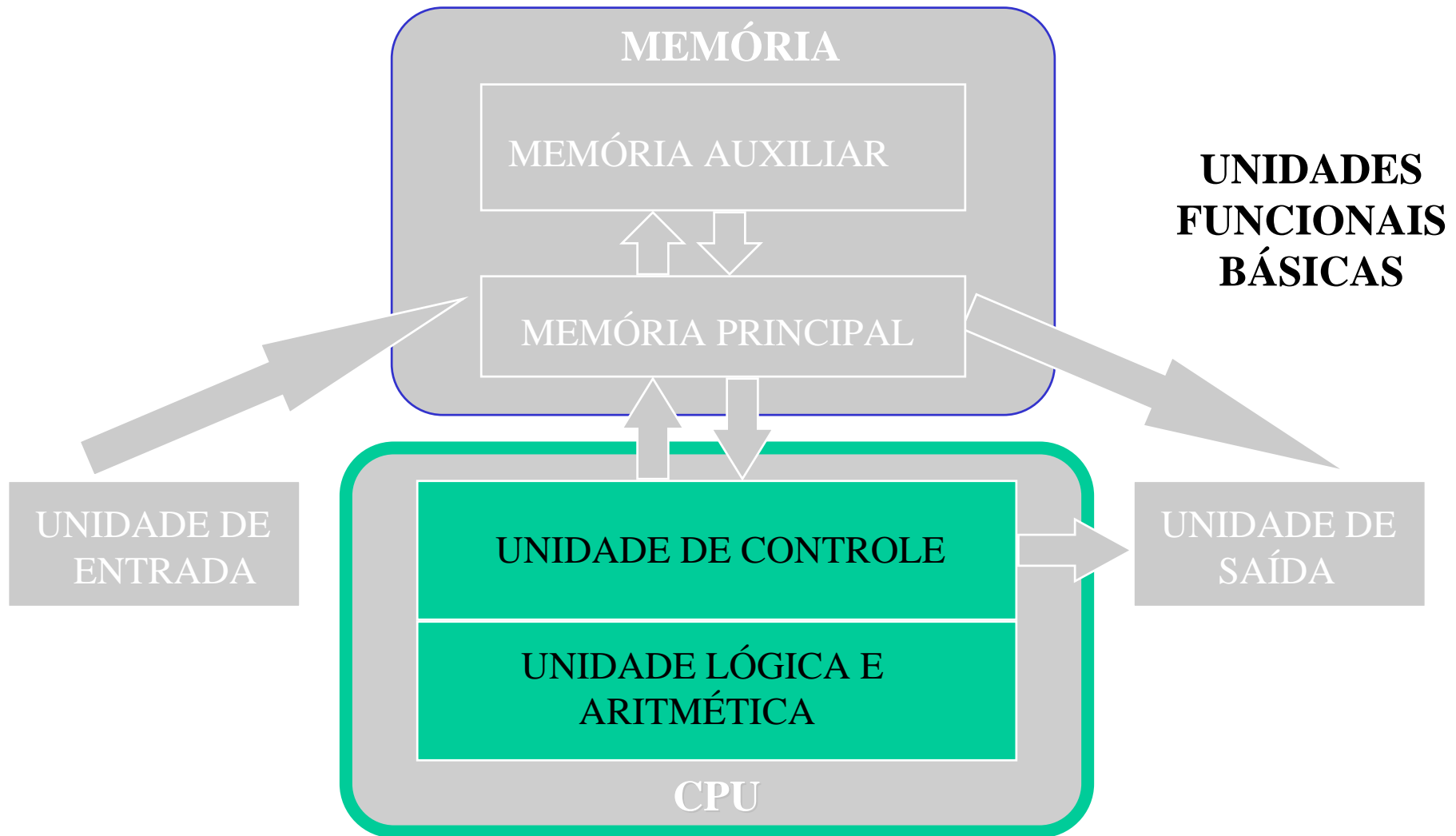
Kbyte = 1024 (2^{10}) bytes

- kilobyte (**Kbyte** ou KB) $\approx 10^3$ bytes
- megabyte(**Mbyte** ou MB) $\approx 10^6$ bytes
- gigabyte(**Gbyte** ou GB) $\approx 10^9$ bytes
- terabyte(**Tbyte** ou TB) $\approx 10^{12}$ bytes

UNIDADES DE ENTRADA E SAÍDA



UNIDADE CENTRAL DE PROCESSAMENTO



UNIDADE CENTRAL DE PROCESSAMENTO

VELOCIDADE DE PROCESSAMENTO: costuma ser expressa através de:

- **MIPS** (milhões de instruções por segundo)

Ex: Pentium de 166 MIPS

- **MFLOPS** ou **MEGAFLOPS** (milhões de operações de ponto flutuante por segundo)

Utilizada em sistemas onde há maior interesse em aplicações numéricas (sistemas científicos e sistemas de computação gráfica)

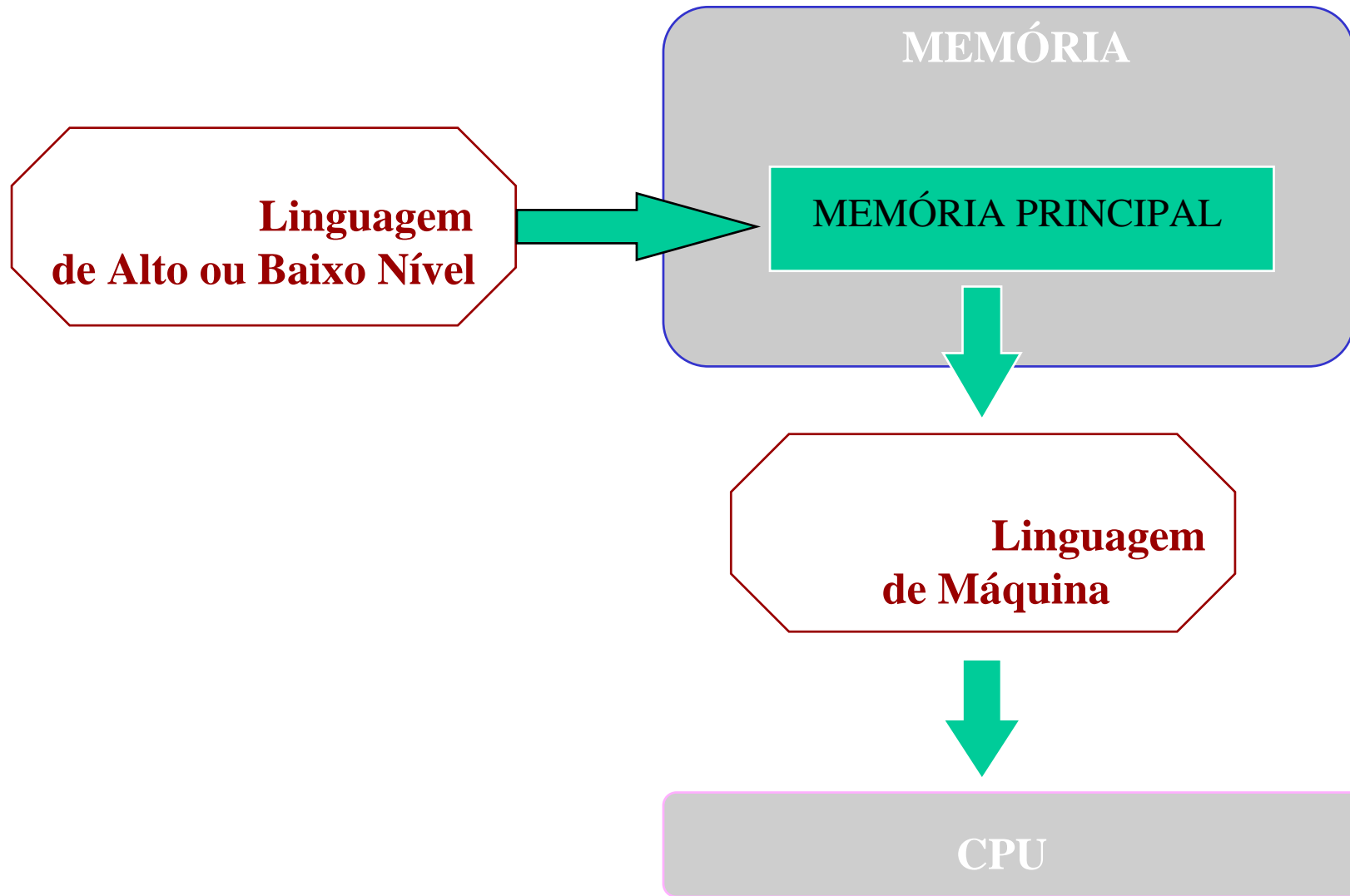
- Comparar com MHz e GHz

O SOFTWARE

O Software envolve um conjunto de:

- 1- INSTRUÇÕES que quando executadas produzem a função e o desempenho desejados
- 2- ESTRUTURAS DE DADOS que possibilitam que os programas manipulem adequadamente a informação
- 3- DOCUMENTOS que descrevem a operação e o uso dos programas

Programação e Níveis de Linguagem



Programação e Níveis de Linguagem

MEMÓRIA

LINGUAGEM DE MÁQUINA

- Uma CPU somente pode compreender instruções que sejam expressas em termos de sua LINGUAGEM DE MÁQUINA
- Um programa escrito em linguagem de máquina consiste de uma série de números binários e é muito difícil de ser entendido pelas pessoas.

Exemplo: Cada instrução é constituída de 2 partes:

código da operação	operando
001	01010

CPU

LINGUAGEM DE BAIXO NÍVEL (Assembly)

São linguagens de programação nas quais os programas são escritos em uma notação que está próxima da linguagem de máquina

Exemplo:

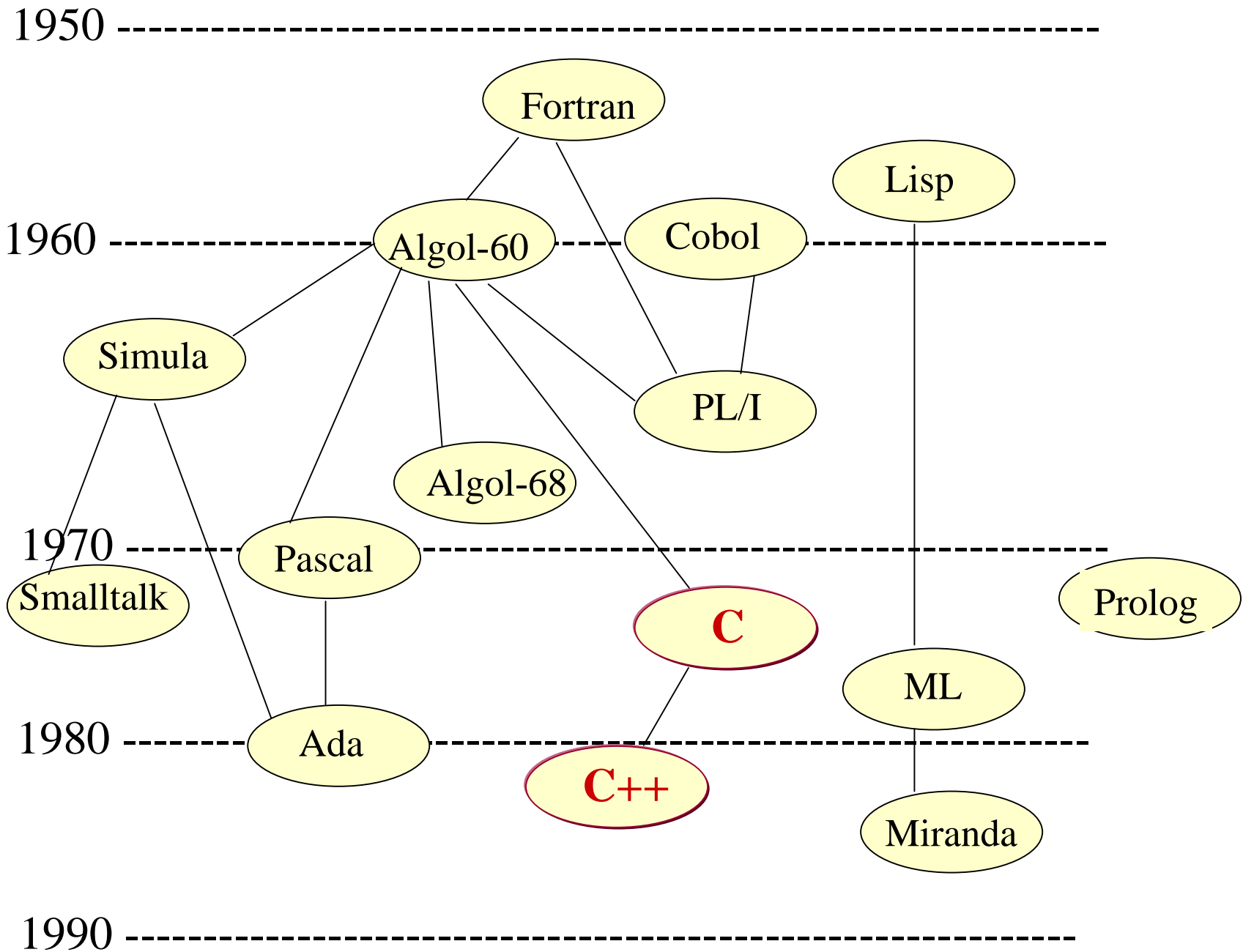
código da operação	operando	significado
LD	A	load A
MPI	5	multiplica 5



CPU

Linguagens de Programação

- Linguagem de Máquina (código binário)
 - inicialmente, programadores trabalhavam diretamente com 1's e 0's: 000110011 = load, ...
- Assembler (simbólico)
 - uso de mnemônicos para as instruções: LD = load
 - labels e variáveis
- Fortran (engenharia, aritmética)
 - linguagem não estruturada
 - ótimos compiladores para aritmética



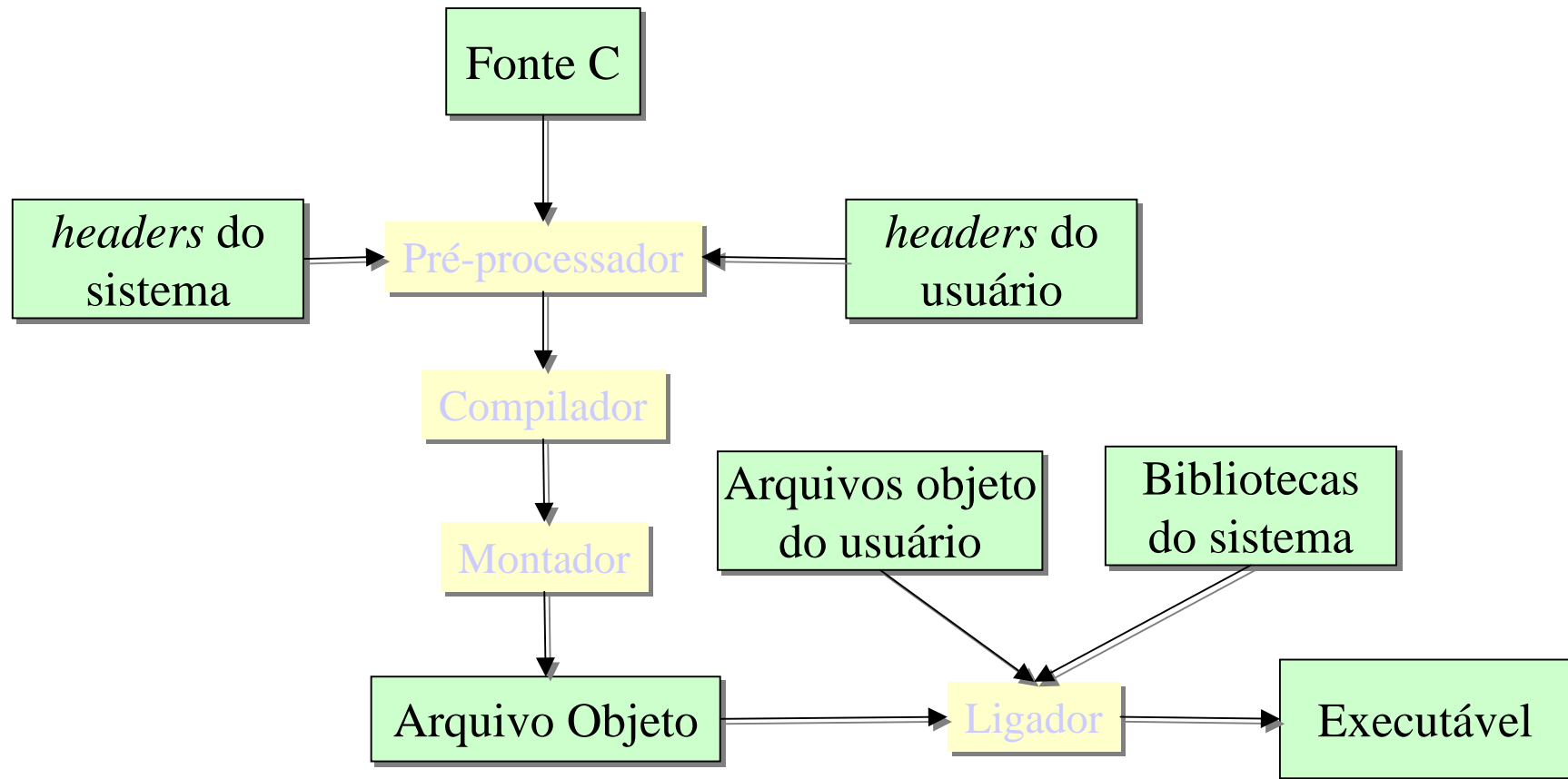
Histórico -”A” Linguagem C

- Origem de C está associada ao sistema Unix
- BCPL: desenvolvida por Martin Richards
- 1970: Ken Thompson desenvolve B, baseada em BCPL, para o primeiro Unix no DEC PDP-7
- 1972: Dennis Ritchie projeta a linguagem C, baseado em B.
 - O sistema operacional Unix, o Compilador C e quase todos os aplicativos Unix **são escritos em C!!!**
- 1988: o *American National Standard Institute* (ANSI) define o padrão ANSI C

Características Gerais

- linguagem de nível *médio*
- não é uma linguagem fortemente tipada
- uso intensivo de ponteiros
- definição de blocos { }
- pré-processador
- não define operações de entrada e saída
- funções retornam valor e podem ser chamadas recursivamente

Fluxo do Compilador C



Arquivo-FONTE

```
/* **** */
/* Primeiro exemplo  arq exemplo1.c */
/* **** */
#include <stdio.h>

/* C padrão de Entrada/Saída */
/* **** */
main () /* Comentários em C */
{
    printf ("exemplo nro %d em C !", 1);
    printf ("\n depois o %d ! \n", 2);
    printf ("criatividade em baixa \n");
}
```

source-file

Compilador

Arquivo-OBJETO

```
1111000101010010
0101100010010100
1110001100010000
1110000000010000
```

object-file

Outros Arquivos
OBJETO/Bibliotecas

```
0101001010000000
1111000101010010
0101100010010100
1110001100010000
1100010100000000
```

libraries

Link-editor

```
1111000101010010
0101100010010100
1110001100010000
1110000000010000
0000000010001010
1100010100000000
0011000100000010
1110000100000011
```

Arquivo-EXECUTÁVEL

Estrutura de um Programa C

Programa C

- Diretivas ao Pré-Processador

- Includes
- Macros

- Declarações Globais

- Funções
- Variáveis

- Definição das Funções

```
main ()  
{ /* begin */  
} /* end */
```

Exemplo

```
#include <stdio.h>
#include <stdlib.h>
#include <conio.h>
#define FALSE 0 /* define F igual 0 */
#define TRUE 1 /* define T igual 1 */

int i = 0;
void mostra_msg( void );

void main( ) {
    int resposta;
    printf( "Quer ver a mensagem?\n" );
    scanf( "%d", &resposta );
    if( resposta == TRUE ) mostra_msg( );
    else puts( "Goodbye for now." );
}

void mostra_msg( void ) {
    clrscr( );
    for( i = 0; i <=10; i++ )
        printf( "Teste # %d.\n", i );
}
```

Uma lista de palavras-chave de C ANSI

(repare, são 32 somente !!!)

auto break case char const
continue default do double else
enum extern float for goto if int
long register return short signed
sizeof static struct switch typedef
union unsigned void volatile while