

MAC2166

Computadores: Arquitetura Básica e Programas em Linguagem de Máquina

Leliane Nunes de Barros
IME-USP

Computador

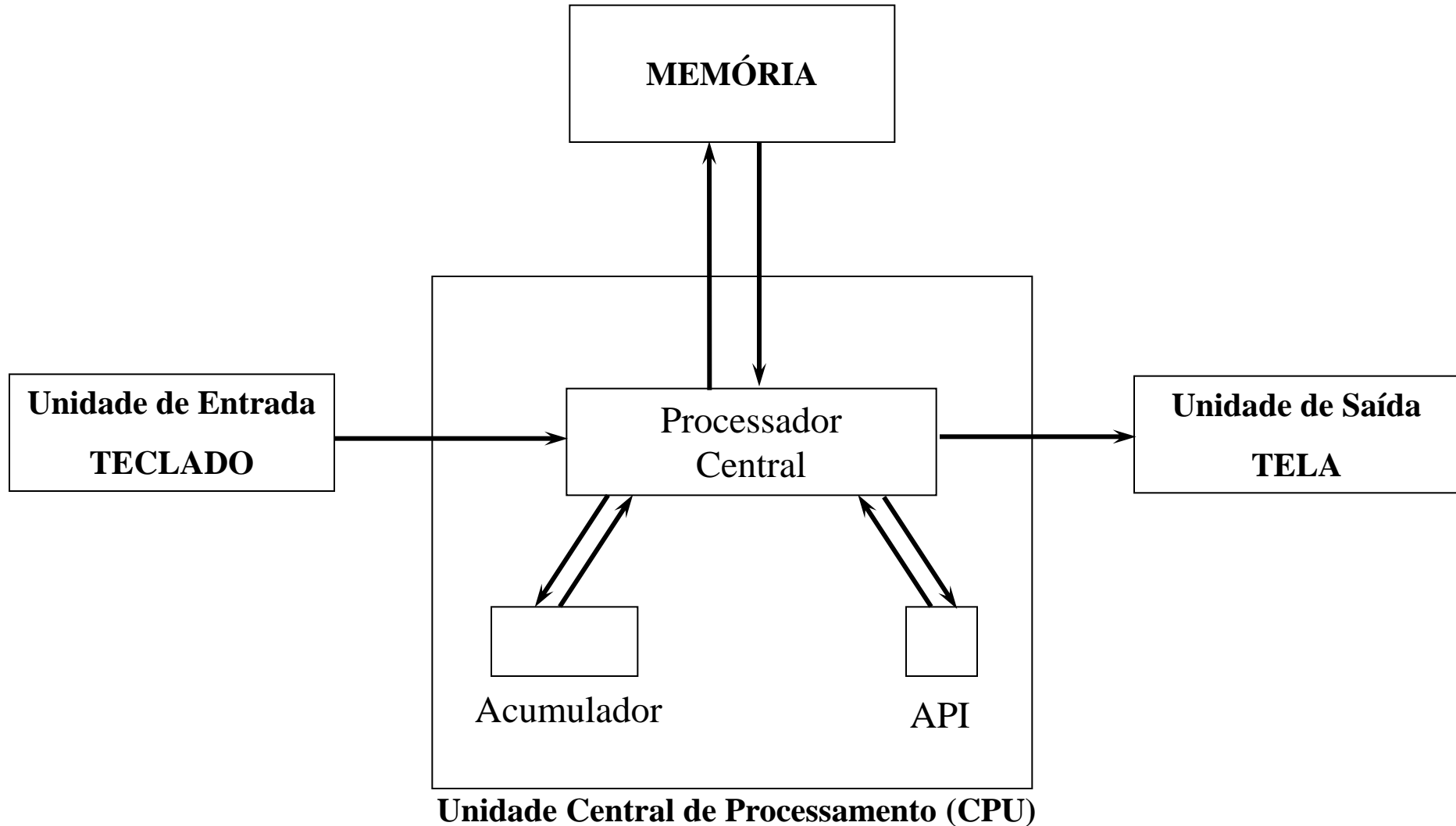
- Serve basicamente para resolver problemas
- Algoritmos: soluções de problemas
- Um programa é uma transcrição de um algoritmo para uma linguagem específica (código) que o computador é capaz de executar
- O programa fica armazenado na memória do computador (diferente da forma que usamos uma calculadora simples)

Exemplo de problema

PROBLEMA 1:

Ler uma seqüência de números inteiros, pelo teclado, e apresentar na tela o resultado da soma da seqüência lida.

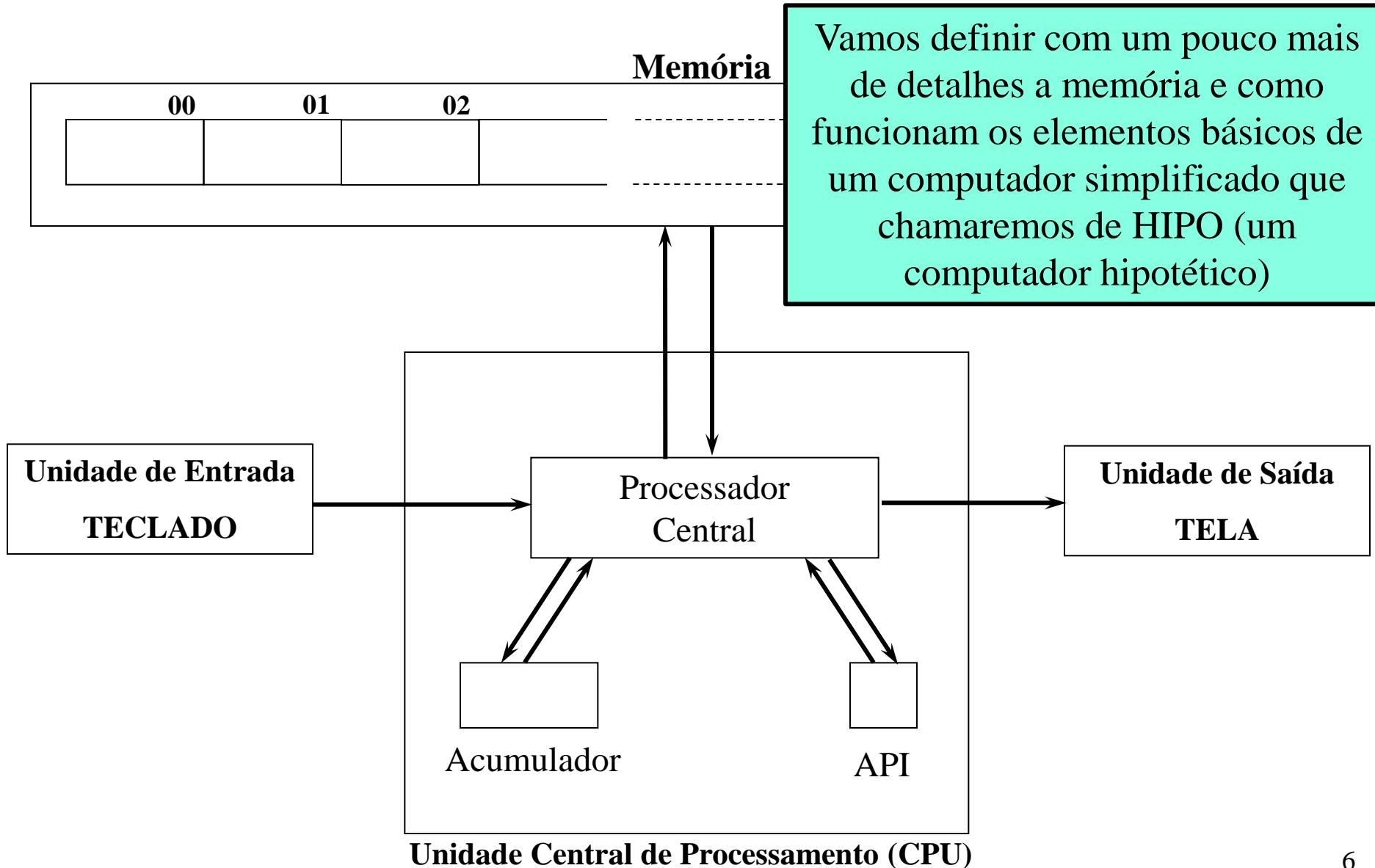
Arquitetura básica de um computador



Arquitetura básica de um computador

- Um programa de computador é composto de instruções básicas
- A CPU é o cérebro do computador ==> onde as instruções são de fato executadas
- *Um cérebro sem memória não serve para nada:* o programa e os dados são armazenados na memória do computador
- O acumulador é um tipo especial de memória da CPU que serve para executar instruções que envolvem 2 operandos
- O API (Apontador de Instruções) serve para apontar a posição de memória onde estará a próxima instrução do programa a ser executada pelo computador

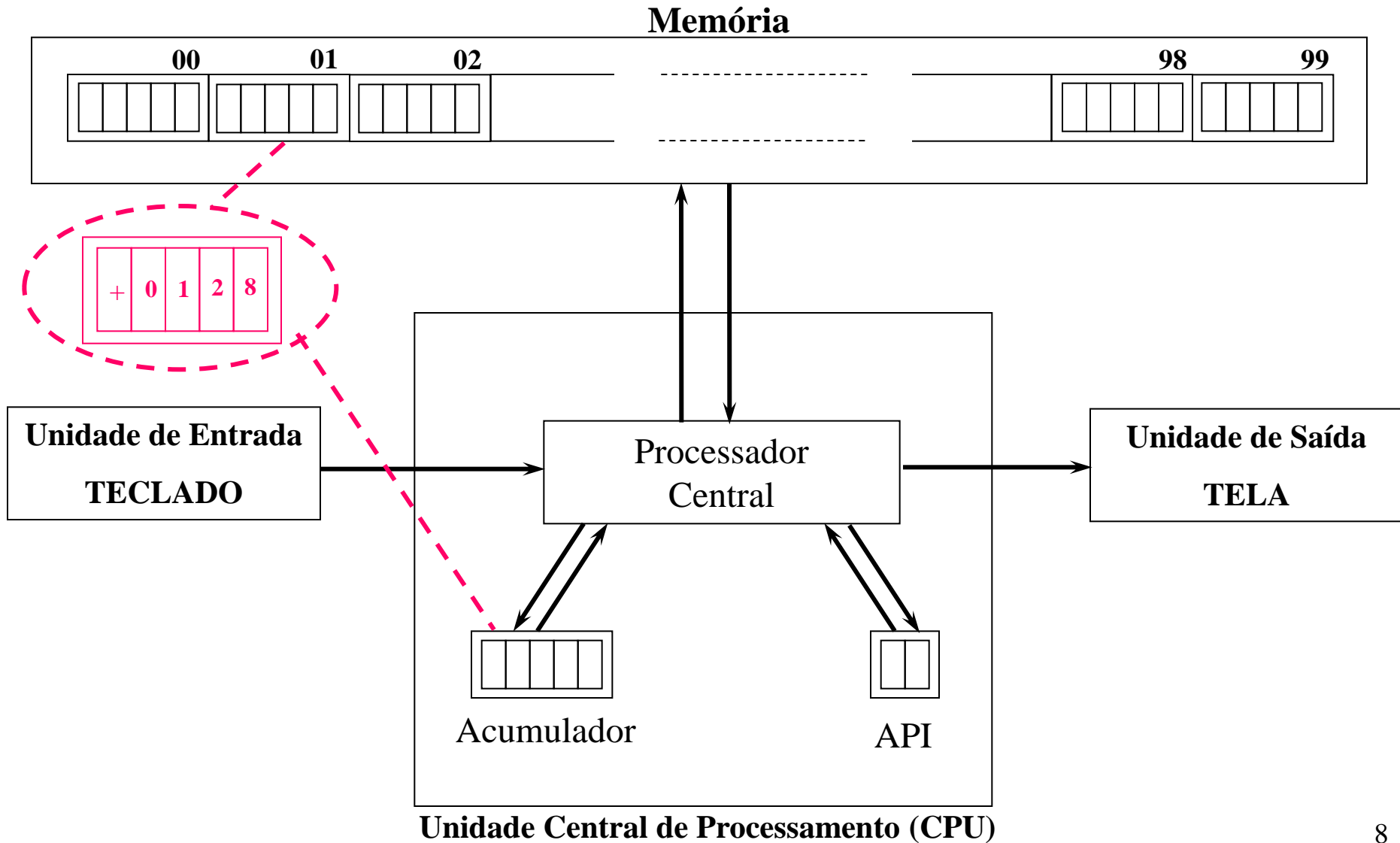
O computador HIPO



Objetivo dessa Aula

- Mostrar como o programa que resolve o Problema 1, pode ser *codificado* na memória do computador HIPO
- Mostrar como o computador HIPO executa o programa usando essa *codificação*

O computador HIPO

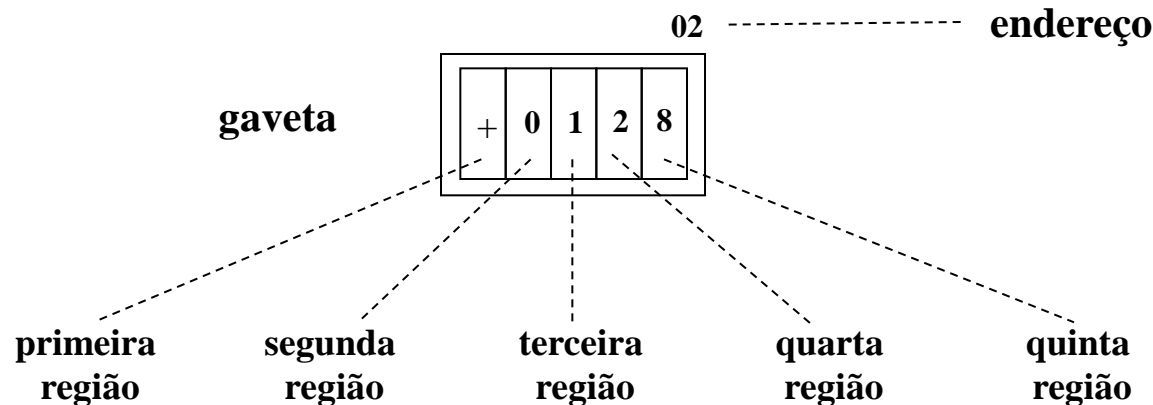


Tamanho de memória

- A memória do nosso computador HIPO, é composta por 100 gavetas enumeradas de 0 à 99
- Dizemos que o tamanho da memória deste computador é 100

Nos computadores reais o tamanho é dado em bytes, por exemplo, 500 Mbytes de memória equivalem a 500 milhões de gavetas (bytes). É comum encontrarmos computadores com tamanho de memória até 8 Gbytes (1 Gbytes = 10^6 bytes).

HIPO: formato da gaveta de memória



- A gaveta do computador HIPO é dividida em 5 regiões
- A primeira região guarda um sinal '+' ou '-'
- Cada região armazena dígitos de 0 a 9 (sistema de enumeração decimal ou de base 10)

Num computador real, cada gaveta (byte) é dividida em 8 regiões (bits) contendo os dígitos 0 ou 1 (sistema binário ou de base 2)

Maior e menor número inteiro

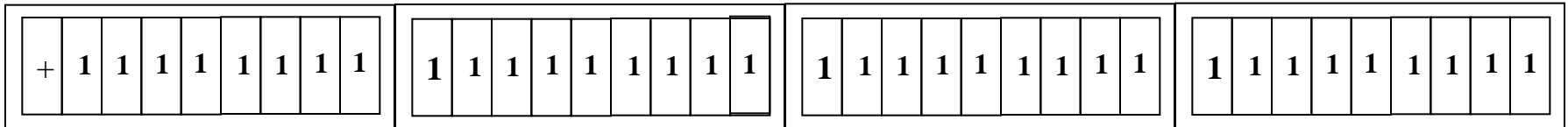
Qual é o maior número que o HIPO pode armazenar?

+	9	9	9	9
---	---	---	---	---

 e o menor?

-	9	9	9	9
---	---	---	---	---

Um computador real, pode usar 4 (ou mais) bytes para armazenar números inteiros. Por exemplo:



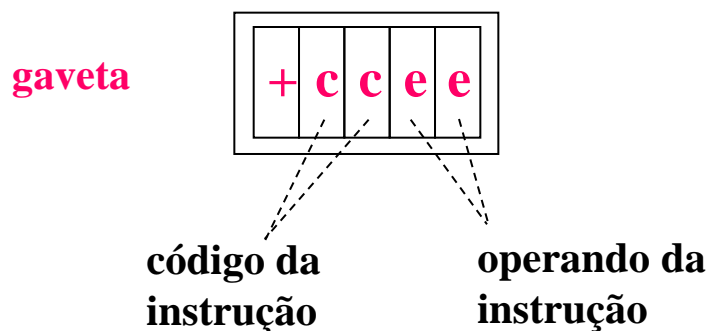
que equivale ao inteiro positivo: $2^{\text{no. de bits}} - 1$

Linguagem de Máquina

- Já sabemos como armazenar números na memória principal
- Como armazenar as instruções do programa?
 - Um computador só pode executar instruções expressas em sua *linguagem de máquina*
 - Vamos definir a linguagem de máquina do computador HIPO
 - Essa linguagem vai ser codificado por números inteiros

A Linguagem de Máquina do HIPO

- Instruções quase sempre estão associadas a uma gaveta de memória
- Formato de uma instrução no HIPO:



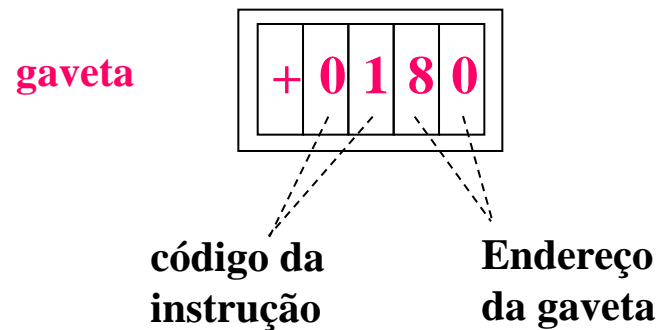
- **cc** são os dois dígitos usados para identificar a instrução, que chamaremos de *código de instrução*
- **ee** são os dois dígitos usados para o endereço de memória associado à instrução, que chamaremos de *operando de instrução*

Instruções do HIPO

Código da instrução	Código Simbólico	Significado da instrução
01	lei	Leia um número do teclado e armazene-o em uma gaveta.
02	imp	Imprima na tela o conteúdo de uma gaveta.
03	cga	Copie o conteúdo de uma gaveta para o acumulador.
04	cag	Copie o conteúdo do acumulador para uma gaveta.
05	som	Some o conteúdo do acumulador com o conteúdo de uma gaveta e guarde o resultado no acumulador.
06	des	Modifique o API para um endereço de uma gaveta.
07	dez	Modifique o API para um endereço de uma gaveta, se o conteúdo do acumulador é igual a zero.
08	fim	Fim das instruções.

Exemplo de uma instrução na memória

Código	Significado
01	Leia um número do teclado e armazene-o na gaveta de endereço 80



Programa: *imprime a soma de dois números lidos*

Execução para as entradas 6 e 4

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80		
81		
82		
83		
...		

API



acumulador



tela

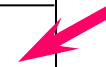


Programa: *imprime a soma de dois números lidos*

Execução para as entradas 6 e 4

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80		
81		
82		
83		
...		

API



acumulador



tela

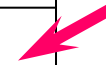


Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e 4

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81		
82		
83		
...		

API



acumulador



tela

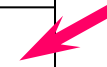


Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83		
...		

API



acumulador

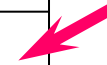
tela

Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83		
...		

API



acumulador

+0006

tela



Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83		
...		

API

acumulador

+0010

tela



Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83	+0010	
...		

API

acumulador

+0010

tela



Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83	+0010	
...		

API

acumulador

+0010

tela

10

Programa: *imprime a soma de dois números lidos*

Execução para as entradas ~~6~~ e ~~4~~

Gaveta	Conteúdo	Símbolo
00	+0180	lei gav
01	+0181	lei gav
02	+0380	cga gav
03	+0581	som gav
04	+0483	cag gav
05	+0283	imp gav
06	+0800	fim
07		
...		
80	+0006	
81	+0004	
82		
83	+0010	
...		

acumulador

+0010

tela

10

Programa: *imprime a soma de dois números lidos*

Execução para as entradas 6 e 4

	Gaveta	Conteúdo	Símbolo
leitura dos dados	00	+0180	lei gav
	01	+0181	lei gav
soma = num1+num2	02	+0380	cga gav
	03	+0581	som gav
	04	+0483	cag gav
	05	+0283	imp gav
	06	+0800	fim
	07		
	...		
num1	80	+0006	
num2	81	+0004	
	82		
soma	83	+0010	
	...		

acumulador
+0010

tela
10

Execução de Programas

No HIPO, a execução de um programa ocorre da seguinte forma:

Passo 1: Carrega o programa na memória, instrução por instrução, a partir do endereço 00

Passo 2: Inicializa o API com 00

Passo 3: Traduz a instrução armazenada apontada pelo API

Passo 4: Aumenta o API de um

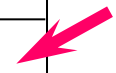
Passo 5: Executa a instrução

Passo 6: Volta para o passo 3

Execute o programa para as seguintes entradas
pelo teclado: 3 , 4 , 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12		
13		

API



acumulador

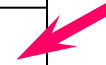


tela



Execute o programa para as seguintes entradas
pelo teclado: 3 , 4 , 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12		
13		



acumulador

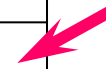
+0000

tela



Execute o programa para as seguintes entradas
pelo teclado: 3 , 4 , 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12		
13	+0000	



acumulador

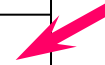
+0000

tela



Execute o programa para as seguintes entradas
pelo teclado: 3 , 4 , 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0000	



acumulador

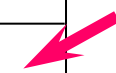
+0000

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0000	



acumulador

+0003

tela



Execute o programa para as seguintes entradas
pelo teclado: ~~3~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0000	



acumulador

+0003

tela



Execute o programa para as seguintes entradas pelo teclado: ~~X~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0000	

acumulador

+0000

tela

Execute o programa para as seguintes entradas
pelo teclado: ~~3~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0000	

acumulador

+0003

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0003	

acumulador

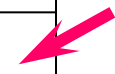
+0003

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0003	
13	+0003	



acumulador

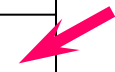
+0003

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0003	



acumulador

+0003

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0003	



acumulador

+0004

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0003	



acumulador

+0004

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0003	

acumulador

+0003

tela

Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0003	

acumulador

+0007

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0007	

acumulador

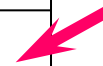
+0007

tela



Execute o programa para as seguintes entradas
pelo teclado: ~~3~~, ~~4~~, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0004	
13	+0007	



acumulador

+0007

tela



Execute o programa para as seguintes entradas
pelo teclado: ~~3~~, ~~4~~, ~~0~~

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	



acumulador

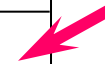
+0007

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, ~~0~~

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	



acumulador

+0000

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, ~~0~~

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	

acumulador

+0000

tela



Execute o programa para as seguintes entradas pelo teclado: ~~3~~, ~~4~~, ~~0~~

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	

acumulador

+0000

tela

7

Execute o programa para as seguintes entradas
pelo teclado: ~~3~~, ~~4~~, ~~0~~

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	

acumulador

+0000

tela

7

O que esse programa faz?

O que esse programa faz?

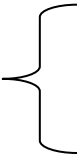
PROBLEMA 2:

Dada uma seqüência de números inteiros, terminada por 0, imprimir sua soma

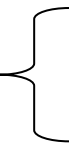
Execute o programa para as seguintes entradas pelo teclado: 3, 4, 0

Gaveta	Conteúdo	Símbolo
00	+0311	cga gav
01	+0413	cag gav
02	+0112	lei gav
03	+0312	cga gav
04	+0709	dez gav
05	+0313	cga gav
06	+0512	som gav
07	+0413	cag gav
08	+0602	des gav
09	+0213	imp gav
10	+0800	fim
11	+0000	
12	+0000	
13	+0007	

inicializa
soma = 0



lê e guarda num
se (num = 0)
desvia para
a gaveta 09



soma recebe
soma + num



desvio incond.

imprime soma



fim

constante = 0

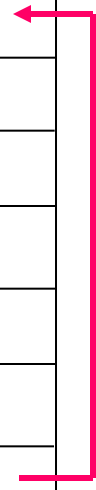
num

soma

seleção



laço ou repetição



acumulador
+0000

tela
7

Linguagem de alto-nível

Programa escrito em linguagem de alto nível (também chamado de *pseudo-código*)

1. soma \leftarrow 0
2. lê e armazena num
3. se (num = 0) desvia para linha 6
4. soma \leftarrow soma + num
5. desvia para linha 2
6. imprime soma
7. fim

A linguagem de máquina do HIPO possui os mesmos recursos básicos computacionais que as linguagens de programação de alto nível possuem

Outros exemplos de computadores hipotéticos:

- **MIX** (Knuth)
- **TOY** (Sedgewick)
- **HIPO** (Setzer)

Compilador

- Um compilador traduz um programa descrito em uma linguagem de programação de alto nível (*programa fonte*) para um programa descrito em linguagem de máquina (*programa executável*), que pode ser diretamente executada pelo computador
- Neste curso, escreveremos programas na linguagem de alto-nível C e usaremos um compilador C para executar os nossos programas em máquinas reais (no computador de casa ou do laboratório)

Principal dificuldade desse curso

- Dado um problema para o qual se conhece uma ou mais soluções (algoritmos), construir um programa que o computador consiga executar , ou seja, traduzir os algoritmos (soluções) que conhecemos para uma linguagem de programação de alto-nível.
 - Algumas vezes, também é difícil criar uma solução (algoritmo) mas como a maioria dos problemas que resolveremos nesse curso possuem soluções simples, o principal desafio será o de traduzir nossos algoritmos para a linguagem C

Por incrível que pareça, existem programas que constróem programas que resolvem problemas. Mas ainda assim é preciso que um ser humano contrua o primeiro programa! Apesar dos computadores serem máquinas incríveis, sempre precisaremos de bons programadores para programá-los.

Lição de casa

PROBLEMA 3:

Escrever um programa que lê um número inteiro e, se for diferente de zero, imprime o número multiplicado por 3

- Desafio extra: observe se o seu programa possui trechos repetidos e, se possível, modifique-o para que ele não tenha partes repetidas.

Desafio

PROBLEMA 4:

Dados um número inteiro n , $n > 0$, e uma seqüência com n números inteiros, escrever um programa que imprima na tela a soma dos inteiros da seqüência.