

INTRODUÇÃO A CONCEITOS DE COMPUTAÇÃO

Software

SUMÁRIO

- ❖ **Software, Programa e Algoritmo**
- ❖ **Tipos de Software**
- ❖ **Algoritmos**
- ❖ **Linguagem de Programação**
- ❖ **Ciclo de Vida do Software**

Algoritmo, Programa e Software

- Um **algoritmo** descreve a sequência de passos de um programa em uma linguagem próxima de um idioma, como o português

Algoritmo para dividir dois valores numéricos

1-Receba o dividendo

2-Receba o divisor

3-Se divisor vale zero

 3.1 Exibir mensagem

 "Divisão por zero não é possível".

 3.2 Terminar a execução.

4. Calcular a divisão:

5. Exibir mensagem "Resultado=divisão"

6. Terminar a execução

Algoritmo, Programa e Software

- Um **programa** é uma sequência de instruções descritas por um algoritmo que permitem com que o computador realize uma tarefa.

```
9  #include <stdio.h>
10 #include<stdio.h>
11 int main()
12 {
13     float dividendo, divisor,divisao;
14
15     printf("Dividendo="); scanf("%f",&dividendo);
16     printf("Divisor="); scanf("%f",&divisor);
17
18     if(divisor==0){
19         printf("Divisão por zero não é possível!!");
20         return 0;
21     }
22     divisao=dividendo/divisor;
23     printf("Resultado=%.2f",divisao);
24     return 0;
25 }
```

```
Dividendo=10
Divisor=0
Divisão por zero não é possível!!
```

```
Dividendo=10
Divisor=3
Resultado=3.33
```

Algoritmo, Programa e Software

- Um **software** é formado por um ou mais programas e estruturas de dados escritos para realizar alguma tarefa que resolva um problema ou torne mais fácil e segura a vida de pessoas ou de organizações.

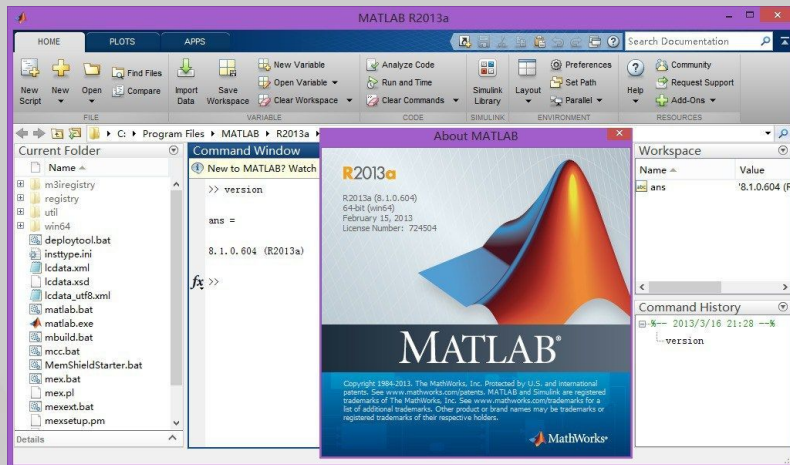


Algoritmo, Programa e Software

Algoritmo

- 1-Receba o dividendo
- 2-Receba o divisor
- 3-Se divisor vale zero
 - 3.1 Exibir mensagem
"Divisão por zero não é possível".
 - 3.2 Terminar a execução.
4. Calcular a divisão:
Divisão= dividendo/divisor
5. Exibir mensagem "Resultado=divisão"
6. Terminar a execução

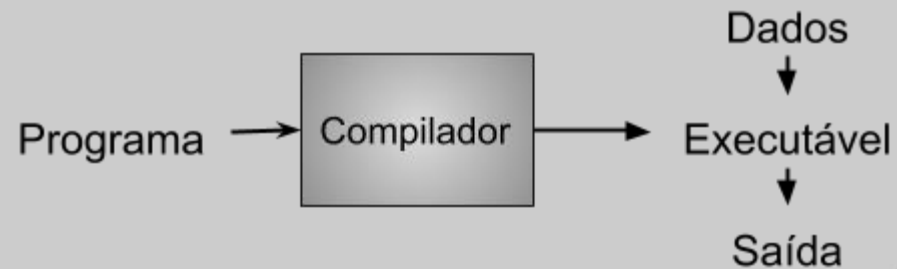
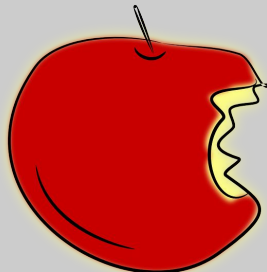
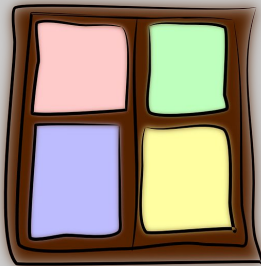
```
9  #include <stdio.h>
10 #include <stdio.h>
11 int main()
12 {
13     float dividendo, divisor, divisao;
14
15     printf("Dividendo="); scanf("%f",&dividendo);
16     printf("Divisor="); scanf("%f",&divisor);
17
18     if(divisor==0){
19         printf("Divisão por zero não é possível!!");
20         return 0;
21     }
22     divisao=dividendo/divisor;
23     printf("Resultado=%.2f",divisao);
24     return 0;
25 }
```



Tipos de Software

- **Software básico ou de sistema:** coleção de programas escritos para apoiar outros programas. Possui contato mais direto com o hardware, gerenciando a execução de outro software.

Sistemas Operacionais Compiladores e Interpretadores



Tipos de Software

- **Software utilitário:** programas úteis ao bom funcionamento do sistema computacional.
- **Software de aplicação:** programas que ajudam o usuário a realizar uma dada tarefa.

Utilitários



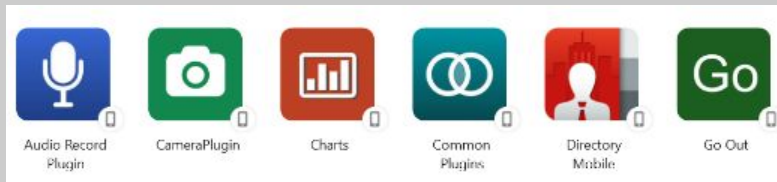
Aplicação



Tipos de Software

- **Plug-in:** código que adiciona características específicas a outro software.
- **Software embarcado:** frequentemente é um firmware dentro de sistemas embarcados ou de dispositivos de uso específico.

Plug-in



Firmware



Algoritmo

- **Conjunto finito de regras, bem definido (sem ambiguidades), voltado à solução de um problema em tempo finito.**
- **Logo, deve possuir 3 qualidades:**
 1. **Cada passo do algoritmo deve ser uma instrução que possa ser realizada**
 2. **A ordem dos passos deve ser precisamente determinada**
 3. **O algoritmo deve ter fim**

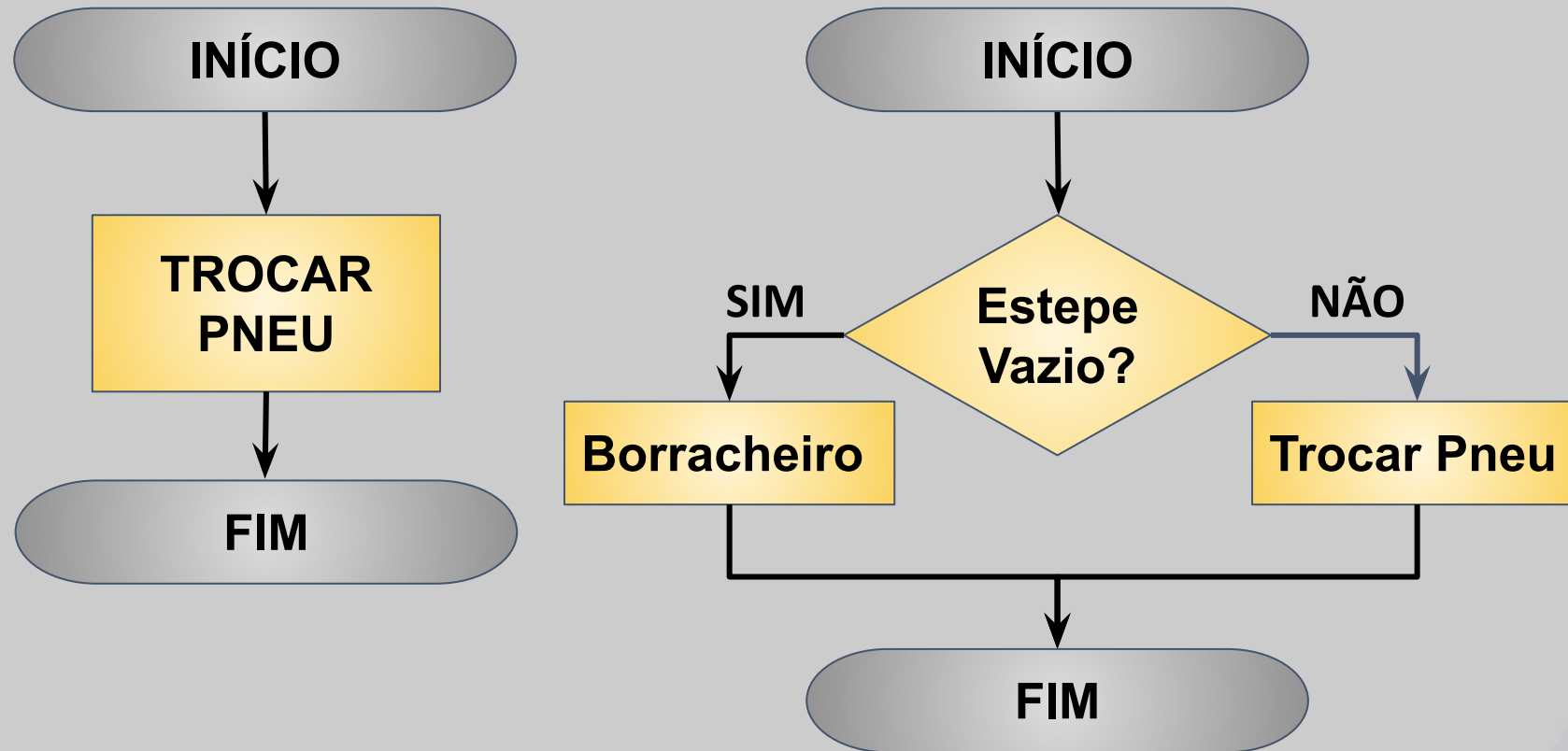
Algoritmo

Dois aspectos devem ser levados em consideração no projeto de algoritmos:

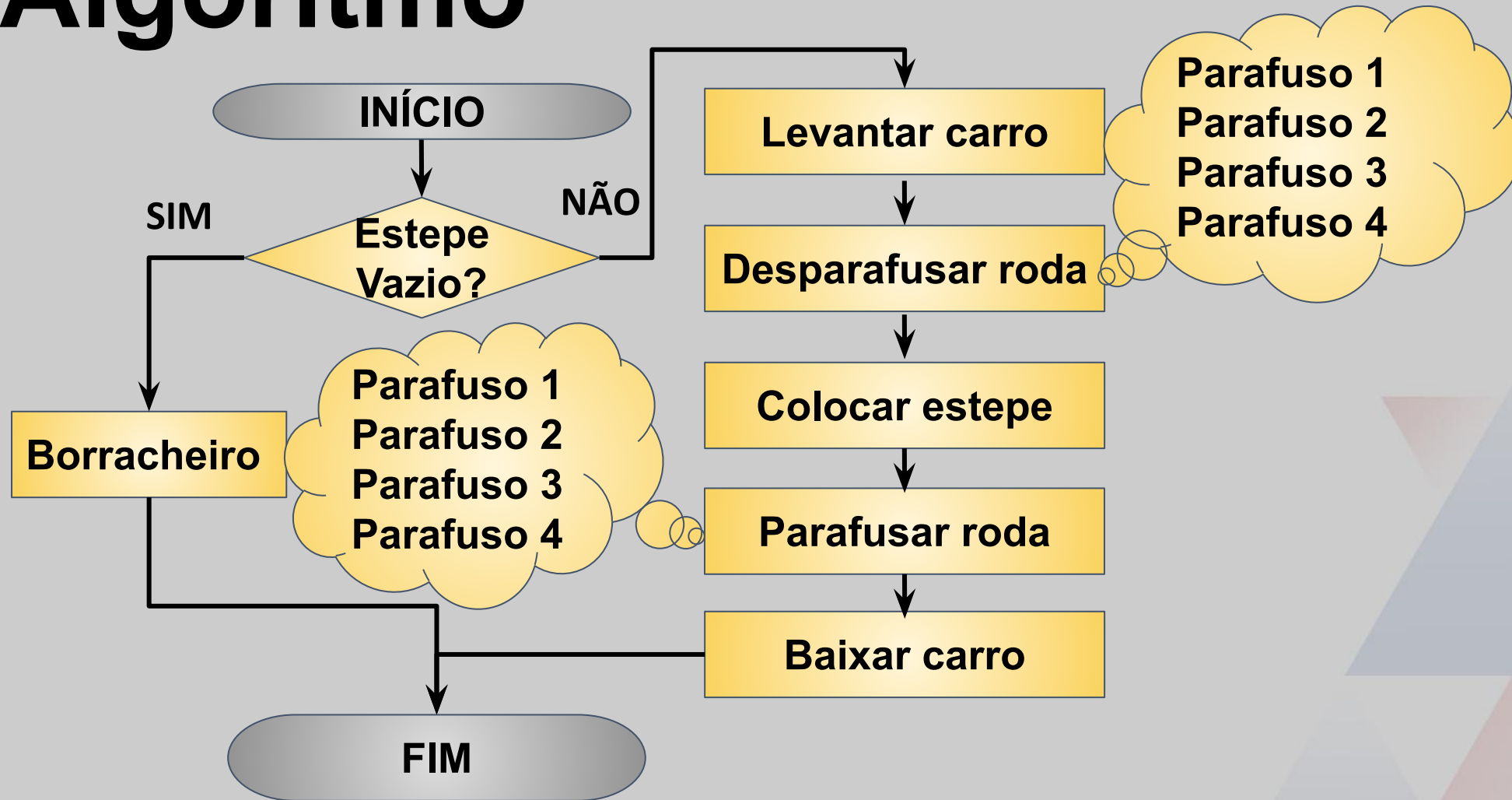
- **Corretude:** para cada entrada possível, o algoritmo terá fim e produzirá a resposta desejada;
- **Eficiência:** a quantidade de recursos (tempo e armazenamento) consumidos na execução do algoritmo.

Algoritmo

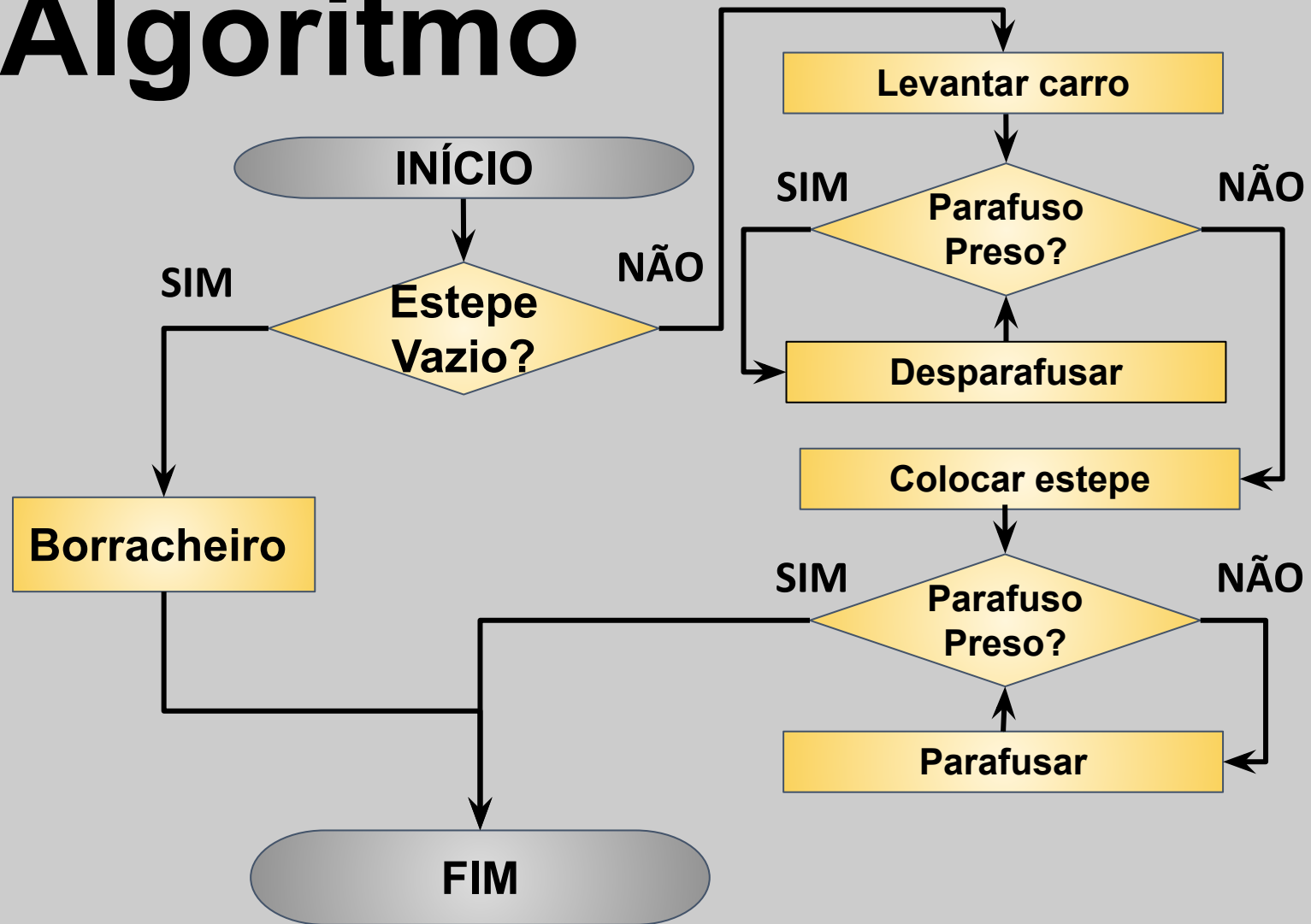
TROCAR PNEU DE UM CARRO



Algoritmo



Algoritmo



Algoritmo

- **Pseudocódigo** é uma forma simples de escrever um algoritmo através de uma linguagem simples.
- Pode seguir algumas regras predefinidas na sua escrita
- Facilita o mapeamento para uma linguagem de programação

Início.

declare n1, n2, soma

escreva("Digite dois números")

leia(n1, n2)

soma = n1 + n2

escreva("Soma é igual a", soma)

Fim.

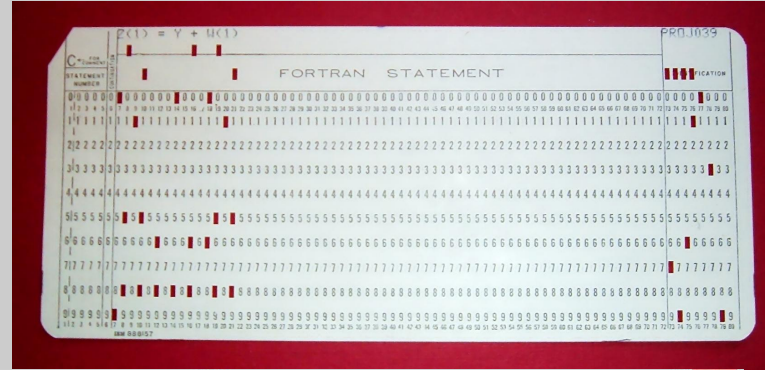
Linguagem de Programação

- **Permite descrever, de forma compreensível para o computador, os passos a serem por ele realizados na execução de uma tarefa.**
- **Toda linguagem de programação possui um conjunto limitado de símbolos, estruturas de dados e comandos utilizados para criar os programas.**



Linguagem de Programação

Linguagens de máquina
Primeira Geração



Linguagens de montagem
Segunda Geração

```
3 section .text
4 global CMAIN
5 CMAIN:
6     mov ebp, esp; for correct debugging
7     call print
8     xor eax, eax
9     ret
10
11 print:
12     ;in eax - odd number, in ecx - even
13     GET_DEC 4, eax
14     cmp eax, 0
15     je .EXIT
16     PRINT_DEC 4, eax
17     PRINT_CHAR ' '
18
19     GET_DEC 4, ecx
20     cmp ecx, 0
21     je .EXIT
22     push ecx
23
24     call print
```

[23:39:58] Build started...
[23:39:58] Built successfully.
[23:39:58] Debugging started..
> p \$eax
\$2 = 5

Linguagem de Programação

Linguagens de alto nível
Terceira Geração

```
9  #include <stdio.h>
10 #include<stdio.h>
11 int main()
12 {
13     float dividendo, divisor,divisao;
14
15     printf("Dividendo="); scanf("%f",&dividendo);
16     printf("Divisor="); scanf("%f",&divisor);
17
18     if(divisor==0){
19         printf("Divisão por zero não é possível!!");
20         return 0;
21     }
22     divisao=dividendo/divisor;
23     printf("Resultado=%.2f",divisao);
24     return 0;
25 }
```

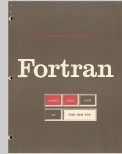
Linguagens de
muito alto nível
Quarta geração



Linguagem de Programação

Paradigmas:

- Imperativo



- Funcional



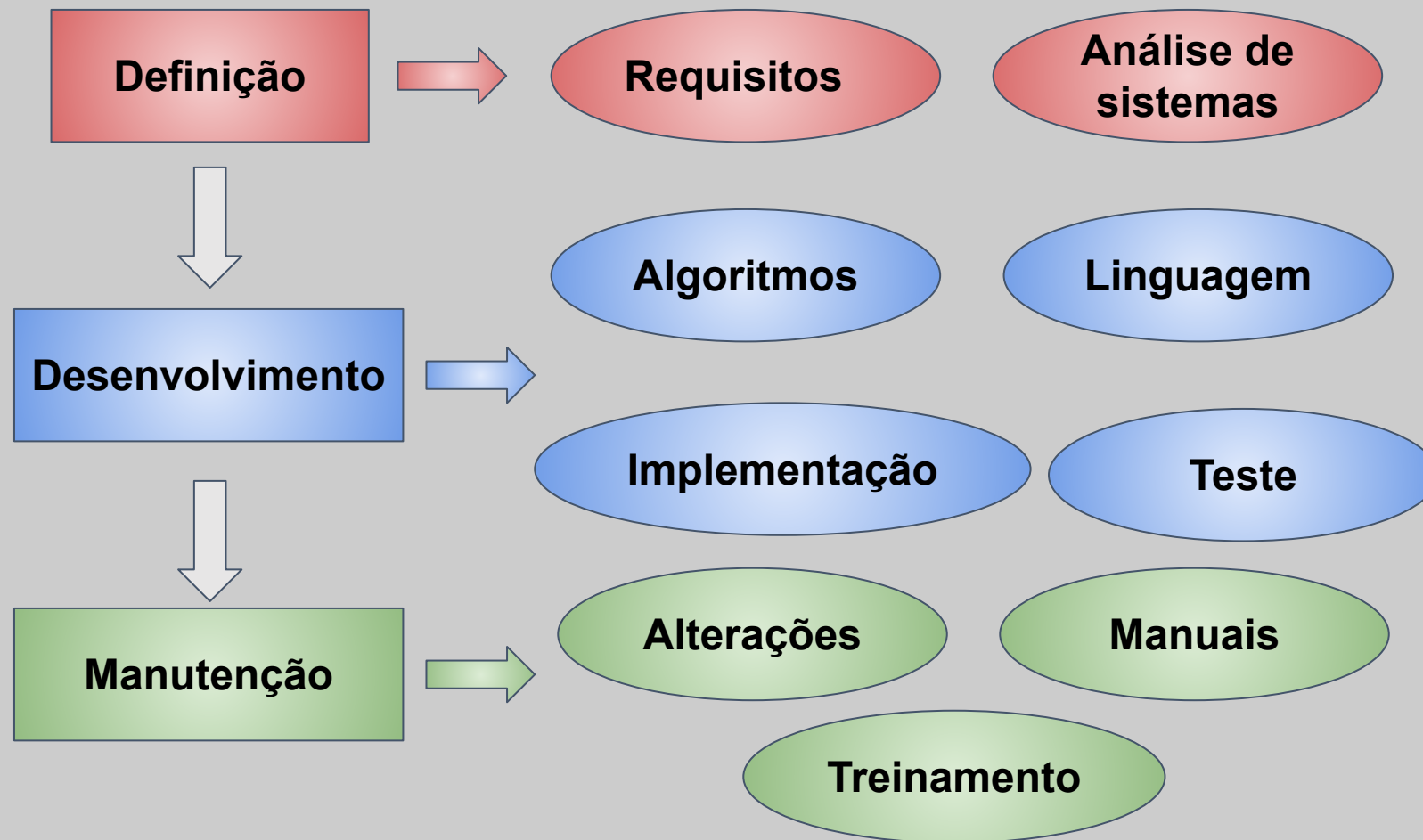
- Lógico



- Orientado a objetos



Ciclo de Vida do Software



INTRODUÇÃO A CONCEITOS DE COMPUTAÇÃO

Software