

Curso de C

Introdução

Copyright © 2005 by Arnaldo V.
Moura e Daniel F. Ferber

Introdução

Roteiro:

- Recordando Algoritmos
- Linguagem de Programação
- O computador
- Instruções de Máquina
- Níveis de Abstração
- Compilação

Algoritmos

Recordando:

Algoritmo: conjunto finito de instruções

- Começa com a primeira instrução
- Execução seqüencial, uma instrução de cada vez
- Possibilidade de saltos para outras instruções
- Sempre alcança a instrução PARE, que termina o algoritmo.

Utiliza dados (**entrada**) e gera um resultado (**saída**)

Algoritmos

Exemplo:

Calcular o máximo divisor comum

Algoritmo em Português

Início

1. → Leia dois números. ←

Entrada

Condição

2. Divida o primeiro pelo segundo e guarde o resto.

3. → Se o resto for 0 (zero), então escreva o segundo número e PARE. ←

Saída

Fim

4. Substitua o primeiro número pelo segundo.

Salto

5. Substitua o segundo número pelo resto da divisão.

6. → Retorne ao passo 2. ←

Algoritmos

Características:

- Cada instrução do algoritmo precisa ser:
 - Não ambígua
 - Uma única operação bem definida
 - Tecnicamente viável
- Necessidade de representação:
 - Diagramas
 - Uma linguagem específica para escrever algoritmos
 - Linguagens de programação

Algoritmos

Exemplo:

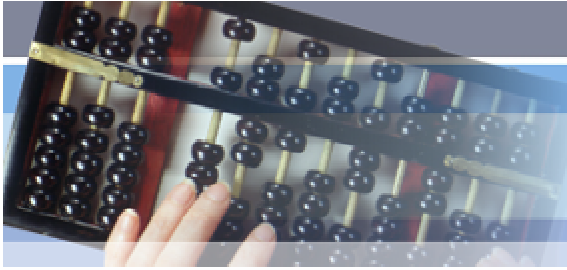
Calcular o máximo divisor comum

Algoritmo em Português

1. Leia dois números.
2. Divida o primeiro pelo segundo e guarde o resto.
3. Se o resto for 0 (zero), então escreva o segundo número e PARE.
4. Substitua o primeiro número pelo segundo.
5. Substitua o segundo número pelo resto da divisão.
6. Retorne ao passo 2.

Potencialmente
ambíguo

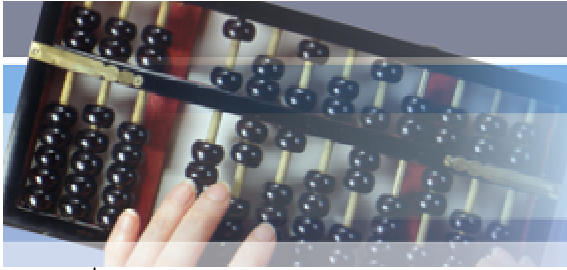
Instrução não está
bem definida



Linguagem de Programação

Opções de representar algoritmos:

- Diagramas
- Uma linguagem específica para escrever algoritmos
- Linguagens de programação



Linguagem de programação

Conceitos:

- A linguagem de programação é um veículo para se escrever algoritmos.
- Características:
 - Vocabulário restrito
 - Regras de sintaxe
 - Recursos automáticos de verificação da sintaxe da linguagem

Exemplo

Exemplo:

Calcular o máximo divisor comum

Linguagem de Programação

Início

Entrada

```
int a, b, r;  
scanf("%d %d", &a, &b);  
do {  
    r = a % b;  
    if (r == 0) break;  
    a = b;  
    b = r;  
};  
printf("O MDC é %d", a);  
return;
```

Palavra chave

Símbolos específicos
de operações

Regra: ponto-e-virgula
após cada operação

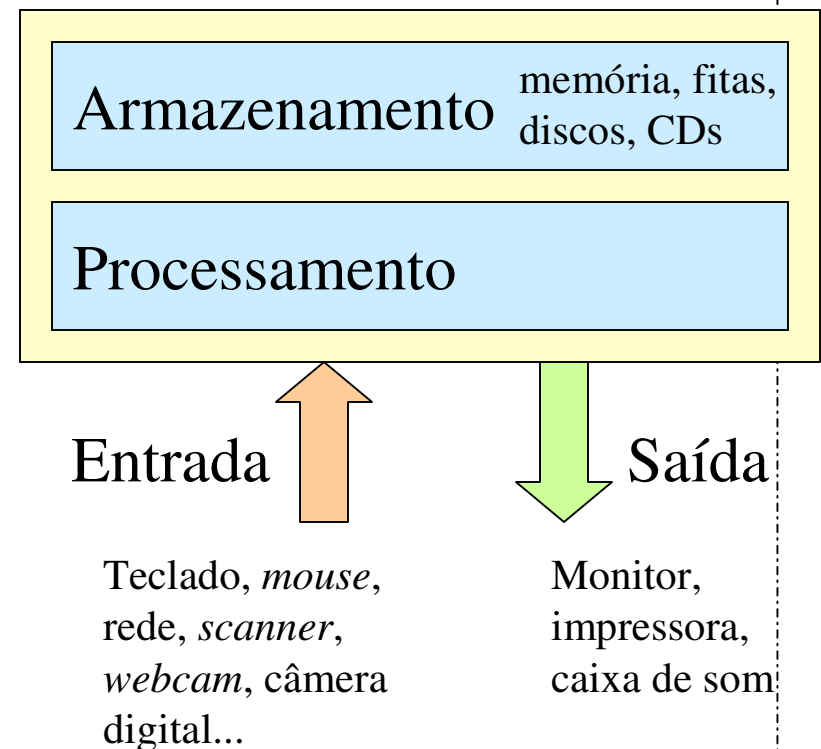
Saída

Fim

O Computador

Conceitos:

- Realiza **processamento** sobre dados **armazenados** no computador.
- Executa **operações matemáticas** e **lógicas** sobre dados.
- Recebe dados do meio externo (**entrada**)
- Apresenta os resultados para o meio externo (**saída**)

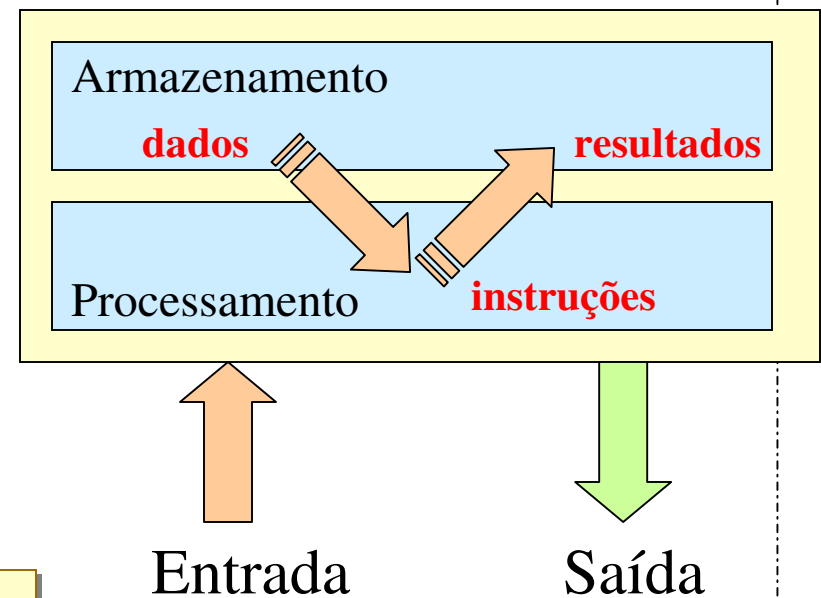


O Computador

Instruções de Máquina:

- Representam operações sobre dados.
- Encontram-se em formato digital (código de máquina)

O computador é uma máquina **rápida** e **eficiente** para **simular algoritmos**!



O Computador

Exemplo: Calcular o máximo divisor comum

Instruções de Máquina

Início

Entrada

Condição

Salto

Saída

Fim

```
IN ax
IN bx
divisao:
MOV dx, 0
DIV bx
CMP dx, 0
JE fim
MOV ax, bx
MOV bx, dx
JMP divisao
fim:
OUT bx
HLT
```

Detalhes do computador:
Controle de circuito eletrônico

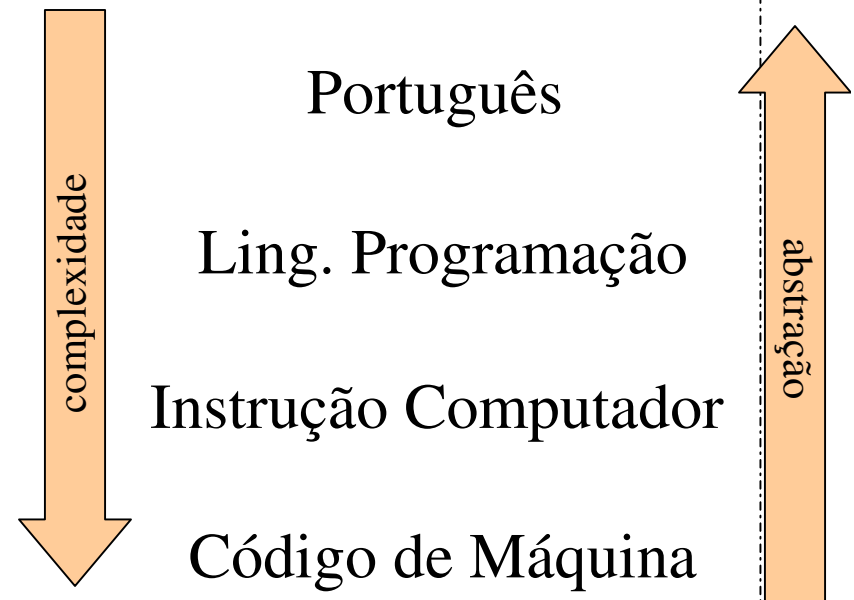
Descrição detalhista

Representação pouco
intuitiva

Níveis de Abstração

Dois casos extremos:

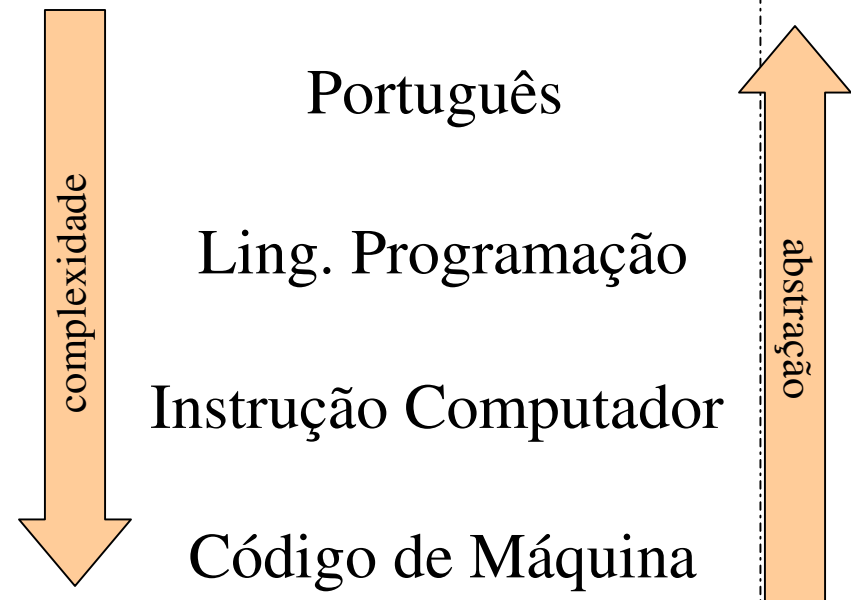
- Português:
 - Fácil, intuitivo
 - Computador não entende
 - Ambíguo, mal definido
- Código de máquina:
 - Complexo e trabalhoso
 - Única forma aceita pelo computador
 - Preciso, bem definido
 - Envolve detalhes específicos do computador, irrelevantes para o algoritmo



Níveis de Abstração

Objetivos:

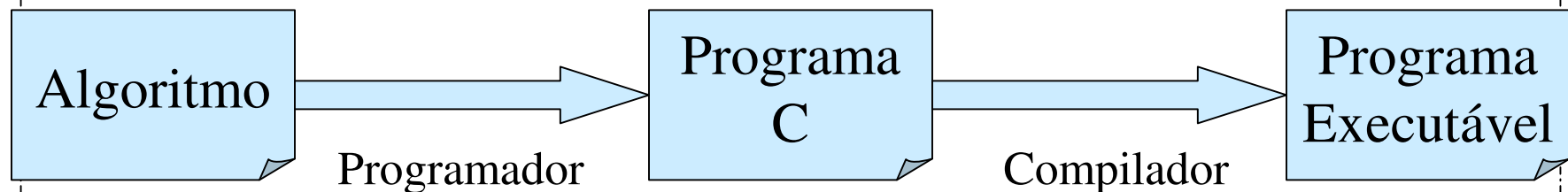
- Descrição precisa do algoritmo
- Independente das características do computador
- Nível de complexidade intermediário!



Construção do Programa

Construir:

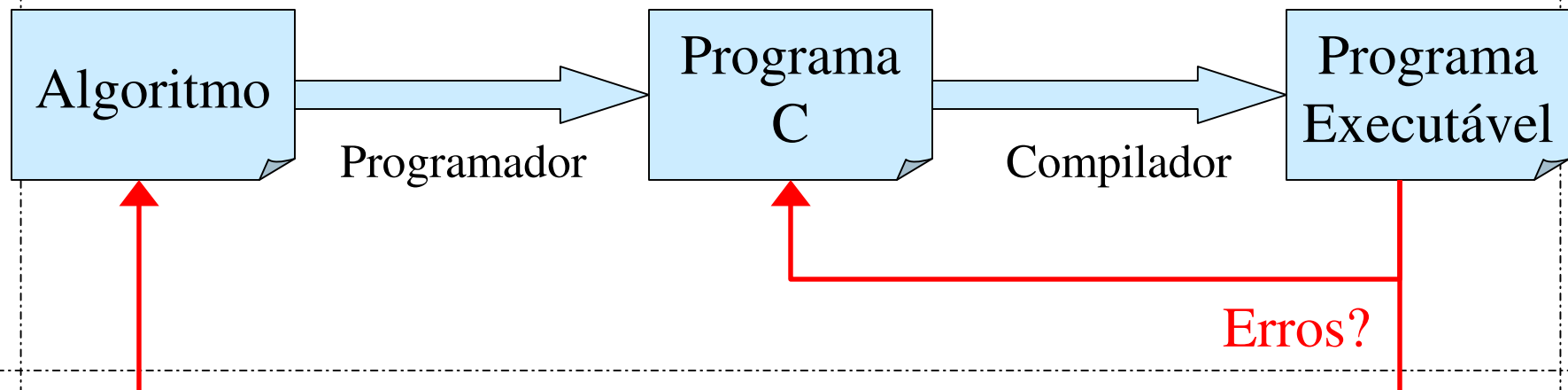
- Passo 1: Elaborar um algoritmo
- Passo 2: Reescrever o algoritmo em C
- Passo 3: Acionar o compilador para gerar o programa executável.



Construção do Programa

Testar:

- Passo 4: Testar o programa
- Erros? Verificar programa em C
- Continuam erros? Verificar o algoritmo!



Introdução

Fim do Capítulo

Copyright © 2005 by Arnaldo V.
Moura e Daniel F. Ferber