



Programação de Computadores:

Introdução a Linguagens de Programação

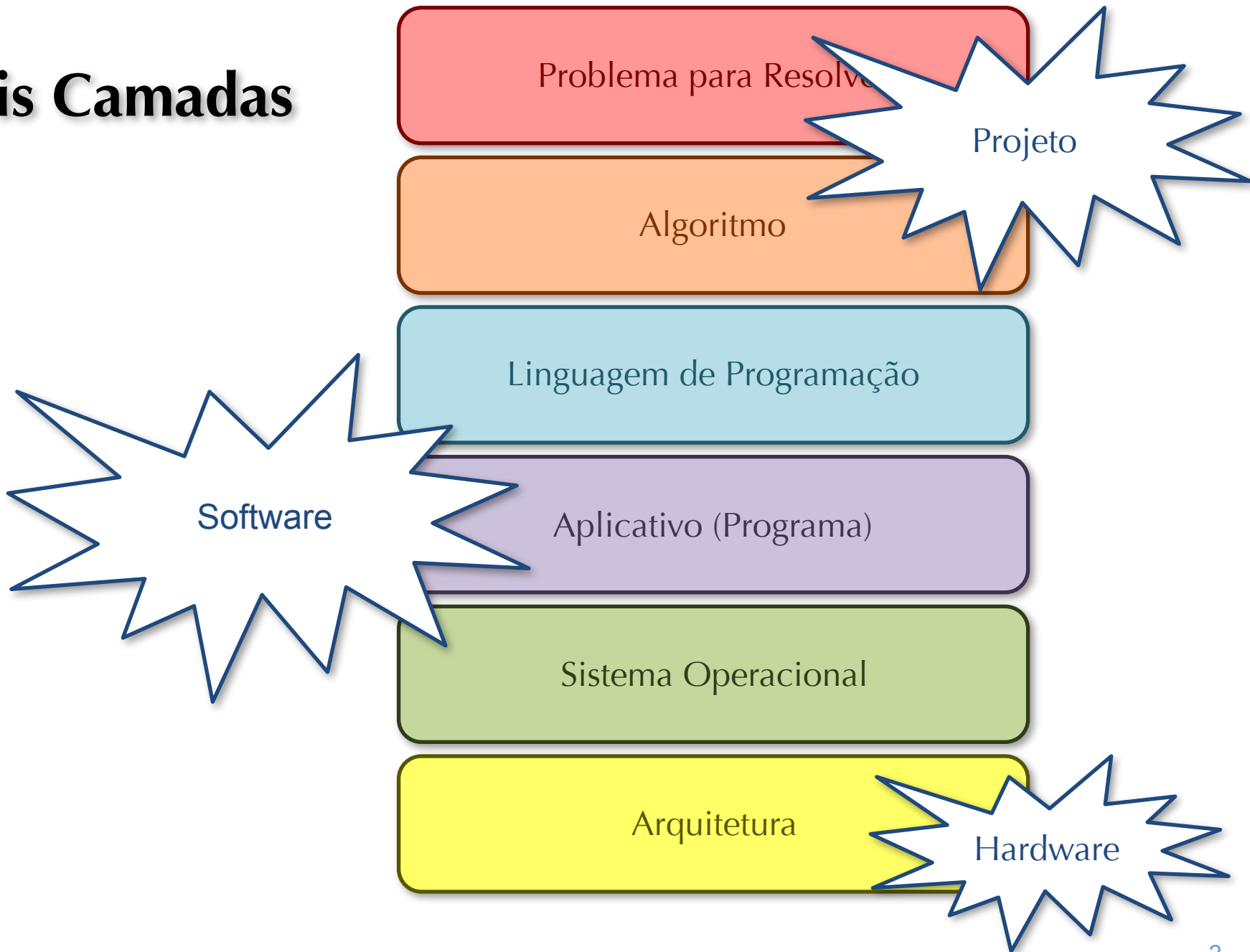
Luis Martí

Instituto de Computação

Universidade Federal Fluminense

lmarti@ic.uff.br - <http://lmarti.com>

Seis Camadas



O que é dado?

- Dado é o elemento a ser processado
- Tipos de dados
 - Numérico
 - 42 (quarenta e dois), 3,14159265... (π), 10^{100} (um googol), etc.
 - Alfabéticos
 - Letras (A-Z)
 - Alfanuméricos
 - Números, letras, caracteres especiais ({, >, +, #)

Processamento de Dados

- É o processo de **receber dados, manipulá-los** e produzir novos dados ou **resultados**



- Exemplos de **processamento manual**
 - Procurar um número telefônico na lista telefônica e anotá-lo numa caderneta
 - Somar valores de compra de supermercado

Computador

- Máquina que **processa dados**
 - Menos tempo
 - Maior segurança contra erro humano
- Funções básicas do computador
 - Entrada de dados
 - Processamento de dados
 - Saída de informações
 - Armazenamento de informações

Dado x Instrução x Programa

- Dado
 - Informação a ser processada
 - Exemplos: Nome do aluno, número de matrícula, notas
- Instrução
 - Operação elementar que o computador pode processar sobre os dados
 - Tipos: transferência de dados, E/S, aritmética, comparação, controle da seqüência do programa, etc.
- Programa
 - Conjunto de instruções, organizadas de forma que o computador as execute em determinada ordem

Dado x Instrução x Programa

Definição do Problema

Leia os cinco números listados abaixo:

3, 5, 2, 4 e -13

Dado

Calcule o somatório desses números e informe o resultado

Programa,
escrito, aqui, em uma
linguagem de programação
hipotética

Solução em um Computador Hipotético

1. Iniciar acumulador com 0 (zero)
2. Ler primeiro valor
3. Acumular valor lido
4. Ler segundo valor
5. Acumular valor lido
6. Ler terceiro valor
7. Acumular valor lido
8. Ler quarto valor
9. Acumular valor lido
10. Ler quinto valor
11. Acumular valor lido
12. Mostrar valor no acumulador

Instrução

Linguagens de Programação

- Método padronizado para **expressar instruções**
- Conjunto de **regras sintáticas** e **semânticas** para definir um programa de computador

Sintaxe

Conjunto de regras formais para a escrita de um programa

Semântica

Diz respeito ao significado do programa sintaticamente válido

Linguagens de Programação

- Método padronizado para **expressar instruções**
- Conjunto de **regras sintáticas** e **semânticas** para definir um programa de computador
- Permite ao programador especificar
 - ... sobre quais dados o computador vai atuar
 - ... como os dados serão armazenados
 - ... como os dados serão transmitidos
 - ... quais ações devem ser tomadas
- Exemplos: Pascal, C, Java, **FORTRAN**, C#

Gerações de Linguagens de Programação

- 1ª Geração – Linguagens de máquina

0010 0001 0110 1100

- 2ª Geração – Linguagens de montagem (*Assembly*)

ADD R1, TOTAL

- ~~• 3ª Geração – Linguagens orientadas ao usuário~~

~~LET SOMA = VAR1 + TOTAL~~

~~IF SOMA > 3 THEN EXIT~~

- 4ª Geração – Linguagens orientadas à aplicação

LIST ALL NOME, ENDERECO, TELEFONE FOR CIDADE = "NITERÓI"

- 5ª Geração – Linguagens de conhecimento (IA)

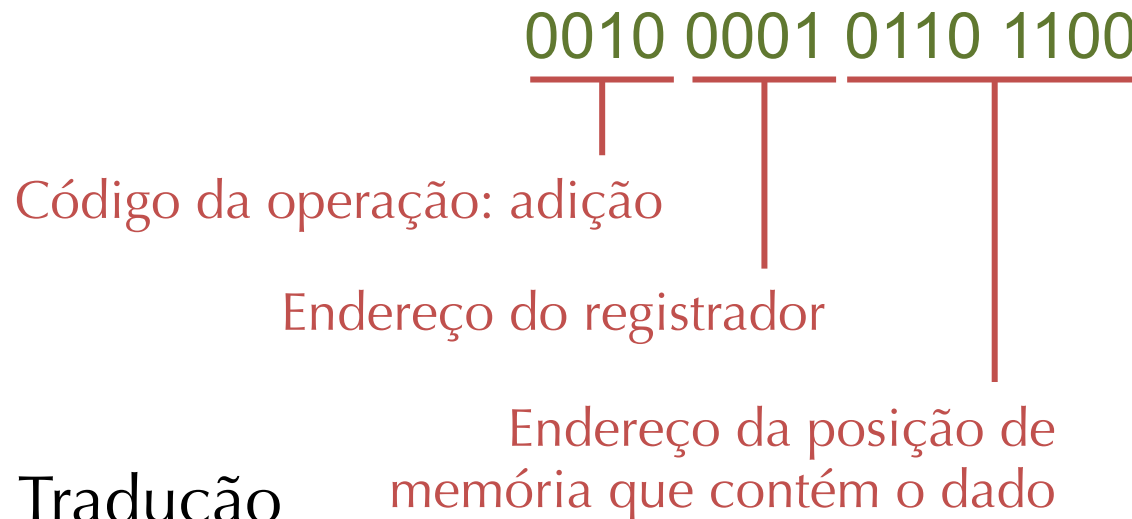
Baixo Nível

Alto Nível

1ª Geração

Linguagens de Máquina

- Linguagem de máquina escrita em notação binária



- Tradução
 - Soma o dado armazenado no registrador 0001 com o dado armazenado na posição de memória 0110 1100

2ª Geração

Linguagens de Montagem

- Linguagem de **baixo nível**
- Elimina a notação binária
 - Introduz o uso de *assemblers* (montadores)
- Usam **códigos mnemônicos** com letras e números para representar os comandos
- Exemplo de comandos
 - **LOAD B**
Carrega no registrador o dado que está no endereço B
 - **ADD A**
Adiciona ao registrador o dado que está no endereço A

3ª Geração

Linguagens Orientadas ao Usuário

- Linguagens de **alto nível**
- Usam comandos com nomes geralmente **auto-explicativos**
 - read, write, if, open, etc.
- Exemplos de linguagens
 - FORTRAN(1954)
 - Cobol (1959)
 - Basic (1965)
 - Pascal (1975)
 - C (1980)

Exemplo FORTRAN

C Primeiro programa de Programacao de Computadores III
C Calcula e imprime a soma de dois valores inteiros

```
PROGRAM Soma2Int
  INTEGER Int1, Int2, Soma
  Int1 = 5
  Int2 = 10
  Soma = Int1 + Int2
  PRINT*, Int1, '+', Int2, '=', Soma
  STOP
END
```

4ª Geração

Linguagens Orientadas à Aplicação

- Geram código a partir de **expressões de alto nível**
- Exemplo de linguagens
 - DBASE, SQL
- Exemplo em DBASE
 - List All Nome, Endereco, Telefone
For Cidade = "Niterói"
- Tradução: lê todos os registros que compõem um arquivo e, para cada registro lido, seleciona aqueles cuja cidade é "Niterói"

5ª Geração

Linguagens de Conhecimento

- Inteligência artificial
- Princípio
 - São criadas bases de conhecimento, obtidas a partir de especialistas, e as linguagens fazem deduções, inferências e tiram conclusões baseadas no conhecimento armazenado
- Exemplos: Prolog, Lisp, Art

Nível das Linguagens de Programação

- Alto Nível
 - Capacidade da linguagem em compreender escritas em **dialetos próximos do inglês** (Ada e Pascal, por exemplo)
- Baixo Nível
 - Aproximam-se da **linguagem própria da máquina**
 - Compostas por **instruções binárias** e outras incompreensíveis para alguém não treinado
- Quanto mais clara para o ser humano (simplicidade maior), mais obscura a linguagem será para a máquina (velocidade menor)

Nível das Linguagens de Programação

Nível Baixo

Nível Médio

Nível Alto



Velocidade

Clareza



Assembler

C

Basic
COBOL
FORTRAN

Pascal

Ada
Modula-2

Montador x Compilador x Interpretador



- Montador
 - Tradutor para linguagens de 2^a geração
- Compilador
 - Traduz linguagens de alto nível, convertendo todo o código fonte em código intermediário ou código objeto
- Interpretador
 - Traduz o programa instrução por instrução

Linguagens de Programação Bizarras

- Top 10 Truly Bizarre Programming Languages

<http://listverse.com/2011/02/17/top-10-truly-bizarre-programming-languages>

Fonte: Listverse

Postado em 17 de fevereiro de 2011

Material adaptado por Luis Martí a partir dos slides de Leandro Augusto Frata Fernandes.