

Paradigmas de Programação

Aula I Apresentação

*Prof.: Rodrigo D Malara
(adaptado de Edilberto M. Silva)*

Ementa

Linguagens de programação: evolução, características; Estruturas sintáticas: notações e gramáticas; Linguagens: imperativas, funcionais, orientadas a objetos, programação lógica; programação concorrente.

Objetivos

- Estudar os princípios de projeto e as características dos principais modelos de linguagens de programação e sua adequação à solução de problemas.*


para que estudá-los?


- Capacidade crescente de expressar idéias.*
- A profundidade que se consegue pensar é influenciada pelo poder da linguagem com a qual se está comunicando os pensamentos.*
- Aumenta-se a habilidade de aprender novas linguagens.*
- Melhora-se a compreensão do significado da implementação.*
- Habilidade maior de projetar novas linguagens.*
- Avanço da computação.*

Conteúdo

Paradigmas de Programação

- 1) Programação Imperativa/Estruturada.
- 2) Programação Orientada a Objetos.
- 3) Programação Funcional.
- 4) Programação em Lógica.

 Estruturada: Fortran, Pascal, “C”, Python

 Orientada a Objetos: Smalltalk, “C++”,
Java, Python

 Funcional: LISP, Clojure, Erlang, Elixir

 Lógica: PROLOG

Linguagens de Programação

O que caracteriza uma linguagem de programação?

- /'\ Sintaxe e semântica bem definidas;*
- /'\ Implementável (executável) com eficiência aceitável.*
- /'\ Universal: deve ser possível expressar todo problema computável.*
- /'\ Natural para expressar soluções de problemas (em um certo domínio de aplicação).*

Por que tantas?

C, C++, Java, Basic, COBOL, Lisp, Haskell, Modula-2, Oberon, Prolog, Java, C#, Pascal, PL/1, Ada, Smalltalk, Símula, Algol, Eiffel, Fortran (1957), ASM, Scheme, CLOS, Maude, Python, Glass, etc.

- “ Propósitos Diferentes;*
- “ Avanços Tecnológicos.*
- “ Interesses comerciais.*
- “ Cultura e background científico.*

Paradigma

- “ O que significa um paradigma?
- “ Como se traduz este significado para o contexto da programação?
- “ Como se pode caracterizar um paradigma?
- “ Paradigma de Programação
 - 🎮 Modelo, padrão ou estilo de programação suportado por linguagens que agrupam certas características comuns.
 - 🎮 A classificação de linguagens em paradigmas é uma consequência de decisões de projeto que têm impacto na forma segundo a qual uma aplicação real é modelada do ponto de vista computacional.

Características Comuns



```
for i:=1 to n do  
begin  
x:=x*i;  
If(x mod 2)=0 then  
writeln(x, 'é par')  
else  
writeln(x, 'é ímpar');  
end;
```

Um problema do mundo real é descrito em termos de variáveis (nome, tipo, endereço), ciclos, condicionais, atribuições, expressões (valor, tipo), entrada e saída, comandos.

O que caracteriza uma Linguagem de Programação?

- “Gramática e significado bem definidos
- Implementável (executável) com eficiência “aceitável”
- Universal: deve ser possível expressar todo problema computável
- Natural para expressar problemas (em um certo domínio de aplicação)

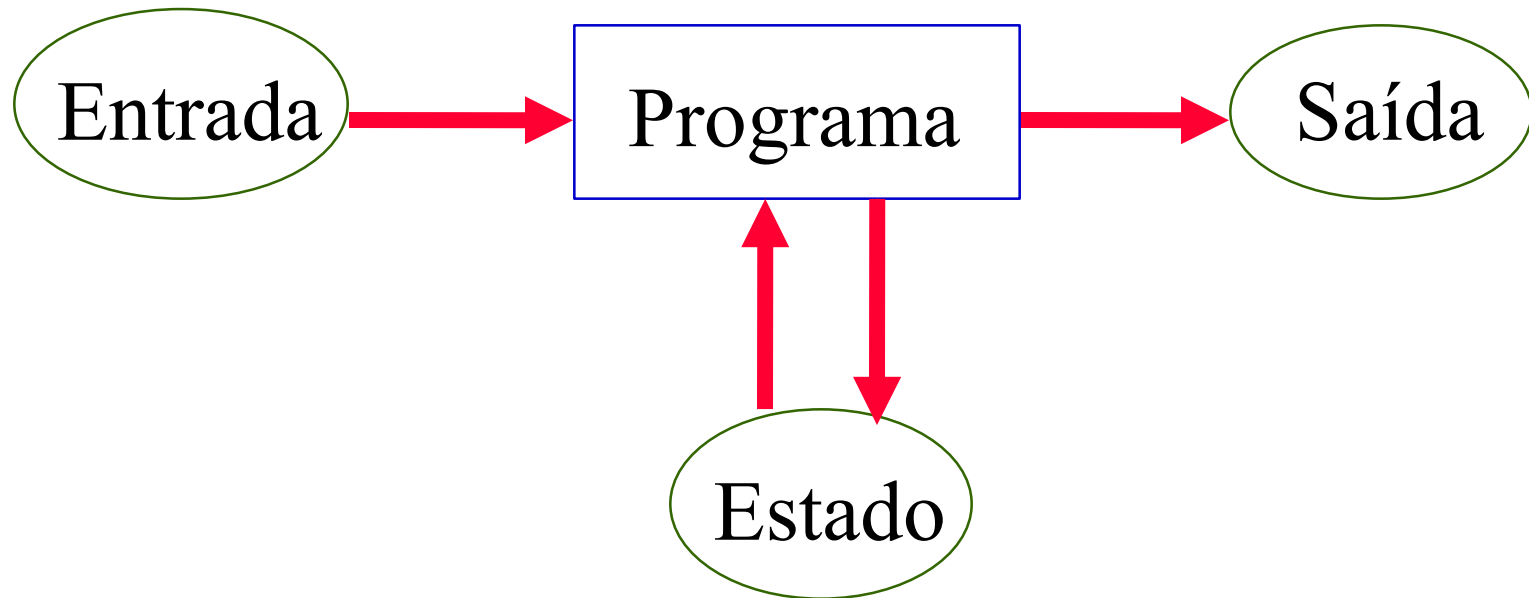
Aspectos do estudo de linguagens

- '\" Sintaxe: gramática (forma)
- '\" Semântica: significado
- '\" Pragmática (ex.: metodologias)
- '\" Processadores:
compiladores, interpretadores, editores,
ambientes visuais ...

O Paradigma Estruturado

- “Programas centrados no conceito de um estado (modelado por variáveis) e ações (comandos) que manipulam o estado
- “Paradigma também denominado de procedural, por incluir subrotinas ou procedimentos como mecanismo de estruturação
- “Primeiro paradigma a surgir e ainda é o dominante

Modelo Computacional do Paradigma Estruturado



Visão Crítica do Paradigma Estruturado

/'\ Vantagens

Eficiência (embute modelo de Von Neumann)

Modelagem “natural” de aplicações do mundo real

Paradigma dominante e bem estabelecido

/'\ Problemas

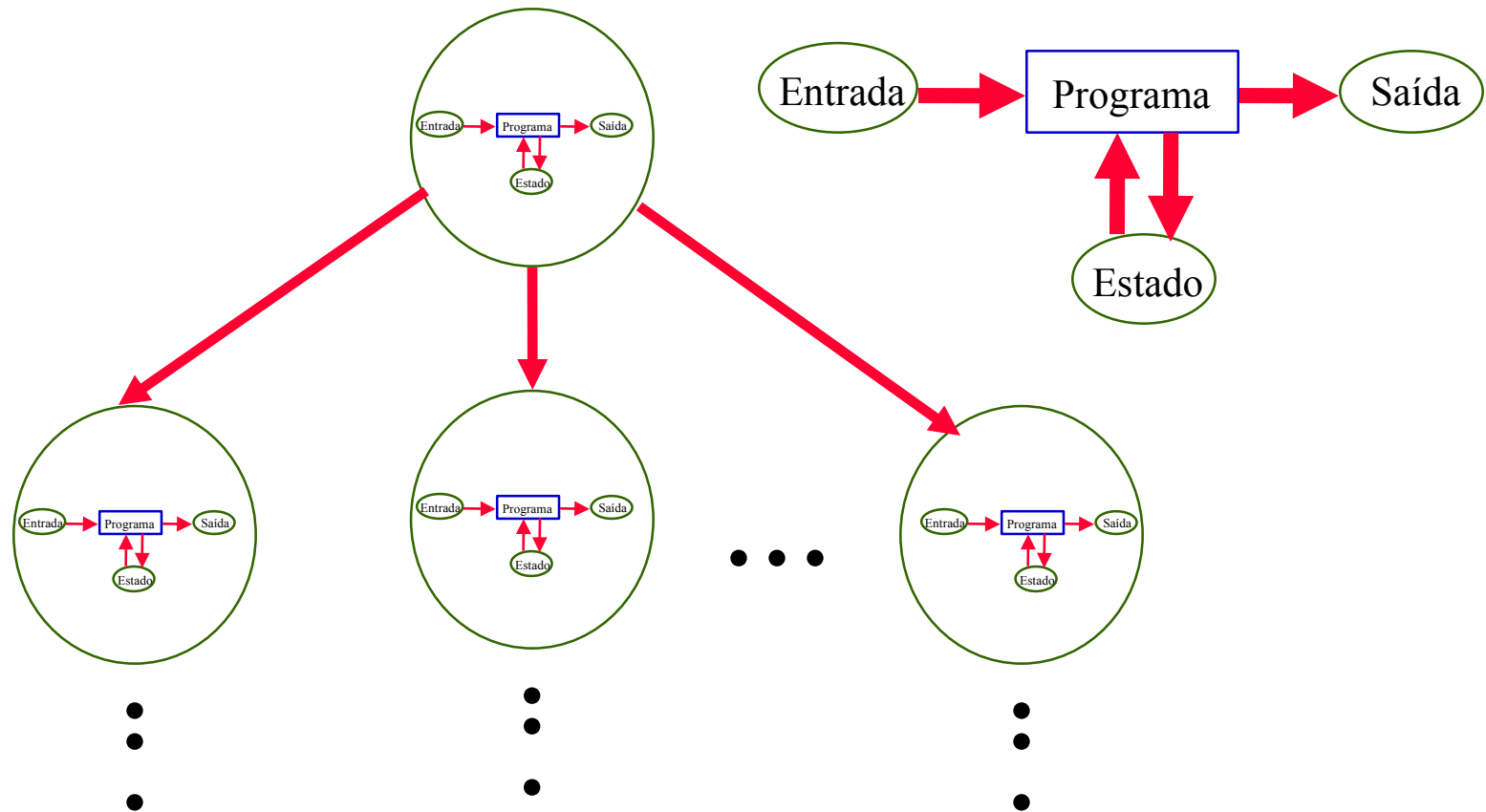
Relacionamento indireto entre E/S resulta em:

- difícil legibilidade
 - erros introduzidos durante manutenção
 - descrições demasiadamente operacionais
- focalizam o como e não o que

O Paradigma Orientado a Objetos

- “ Não é um paradigma no sentido estrito: é uma extensão do estruturado
- “ A diferença é mais de metodologia quanto à concepção e modelagem do sistema
- “ A grosso modo, uma aplicação é estruturada em módulos (**classes**) que agrupam um estado e operações (**métodos**) sobre este
- “ Classes podem ser estendidas e/ou usadas como tipos (cujos elementos são **objetos**)

Modelo Computacional do Paradigma Orientado a Objetos



Visão Crítica do Paradigma Orientado a objetos

/' **Vantagens**

Todasas do estilo estruturado

Classes estimulam projeto centrado em dados:
modularidade, reusabilidade e extensibilidade

Aceitação comercial crescente

/' **Problemas**

Semelhantes às do paradigma estruturado, mas
amenizadas pelas facilidades de estruturação

O Paradigma Funcional

- “ Programas são funções que descrevem uma relação explícita e precisa entre E/S
- “ Estilo declarativo: não há o conceito de estado nem comandos como atribuição
- “ Conceitos sofisticados como polimorfismo, funções de alta ordem e avaliação sob demanda
- “ Aplicação: prototipação em geral e IA

Modelo Computacional do Paradigma Funcional



Visão Crítica do Paradigma Funcional

Vantagens

Manipulação de programas mais simples:

- Prova de propriedades
- Transformação (exemplo: otimização)

Concorrência explorada de forma natural

Problemas

“O mundo não é funcional!”

Implementações ineficientes

Mecanismos primitivos de E/S e formatação

O Paradigma Lógico

- “Programas são relações entre E/S
- “Estilo declarativo, como no paradigma funcional
- “Na prática, inclui características imperativas, por questão de eficiência
- “Aplicações: sistemas especialistas e banco de dados

Modelo Computacional do Paradigma Lógico



Visão Crítica do Paradigma Lógico

Vantagens

Em princípio, todas do paradigma funcional

Permite concepção da aplicação em um alto nível de abstração (através de associações entre E/S)

Problemas

Em princípio, todos do paradigma funcional

Linguagens usualmente não possuem tipos, nem são de alta ordem

Tendência: integração de paradigmas

- “ A principal vantagem é combinar facilidades de mais de um paradigma, aumentando o domínio de aplicação da linguagem
- “ Exemplos: linguagens lógicas ou funcionais com o conceito de estado e comandos
- “ A integração deve ser conduzida com muita cautela, para que não se viole os princípios básicos de cada paradigma.

Outras Classificações

- “ Linguagens de 1a., 2a., 3a. 4a. e 5a. gerações
- “ Programação seqüencial versus concorrente
- “ Programação linear versus programação visual
(*visual programming*) Ex.: Visual
BASIC, Delphi, Embarcadero
- “ ...

Referências Bibliográficas

BIBLIOGRAFIA BÁSICA

- “ SEBESTA, R. W. Conceitos de Linguagens de Programação. Porto Alegre: Bookman, 2000.
- “ MIZRAHI, V. V. Treinamento em Linguagen C++. Módulo 2. 2º Edição. São Paulo: Pearson Prentice Hall, 1994.

BIBLIOGRAFIA COMPLEMENTAR

- “ GOMES, A. F.; VENERUCHI E. A. Fundamentos da Programação de Computadores. São Paulo: Pearson Education do Brasil, 2003.