

Paradigmas de Programação

Juliano Mengue¹, Rodrigo Casara¹

¹Curso de Sistemas de Informação - Faculdades Integradas de Taquara (FACCAT)

bananastaut@gmail.com, rodrigocasara@bol.com.br

***Resumo.** Um paradigma de programação fornece a visão que o programador possui sobre a estruturação e execução do programa. Assim como diferentes grupos em engenharia de software propõem diferentes metodologias, diferentes linguagens de programação propõem diferentes paradigmas de programação. Algumas linguagens foram desenvolvidas para suportar um paradigma específico, enquanto outras suportam múltiplos paradigmas. Paradigma de programação é a forma como o programador enxerga a solução do problema. Em outras palavras, paradigma é a metodologia de solução.*

1. Programação Imperativa

Descreve a computação como ações que mudam o estado de um programa, programas imperativos são uma sequência de comandos para o computador executar. São muito parecidos com o comportamento das linguagens naturais que expressam ordens.

Exemplos: Basic, C, Fortran, Pascal, Python.

2. Programação Estruturada

É uma forma de preconizar todos os programas possíveis reduzindo à apenas três estruturas: sequência, decisão e iteração. Programação estruturada orienta os programadores para a criação de estruturas simples em seus programas, usando as sub-rotinas e as funções. Pode-se dizer ainda que é marcadamente influente, uma vez que grande parte das pessoas ainda aprende programação através dela.

Muitas das linguagens imperativas são também estruturadas, exemplos: Pascal, C, Python...

3. Programação Orientada a Objetos

É um paradigma de análise, projeto e programação de sistemas baseado na composição e interação entre diversas unidades de software chamadas de objetos. Orientação a objetos têm como definição identificar o melhor conjunto de objetos para descrever um sistema de software, que se faz relacionando a troca de mensagens entre estes objetos.

Com um conjunto de Classes (Que definem os objetos presentes no sistema de software) determinamos o comportamento e estados possíveis de seus objetos, assim como o relacionamento com outros objetos.

Exemplos de linguagens orientadas a objetos: C++, Java, Object Pascal, Ruby.

4. Programação Funcional

Trata a computação como uma avaliação de funções matemáticas e que evita estados ou dados mutáveis. Ela enfatiza a aplicação de funções, em contraste da programação imperativa, que enfatiza mudanças no estado do programa.

À programas que usam desse artifício (funções), mas não quer dizer que são linguagens funcionais (ex.: Pascal).

Exemplo de linguagem funcional: Linguagens puramente funcionais tem sido mais usadas academicamente, as mais usadas na indústria e comércio são Erlang, R, Mathematica, J e K, XSLT.

5. Programação Lógica

Programação que faz uso da lógica matemática como fundamento, o sentido da programação lógica é trazer o estilo da lógica matemática à programação de computadores.

A lógica proporciona uma maneira de demonstrar se uma questão é verdadeira ou falsa, o programa básico formará conclusões imediatas a partir de uma lista de premissas.

Essas conclusões serão tanto sentenças declarativas quanto imperativas. Quando uma sentença imperativa é deduzida, o programa toma uma ação correspondente, esse paradigma de programação é muito usado no desenvolvimento de software de I.A. (Inteligência Artificial).

O exemplo de linguagem Lógica mais conhecido é o PROLOG, criado em 1972 por Alain Colmerauer, a primeira linguagem a usar esse método foi a Planner, mas temos ainda Qa-4, Popler, Qlisp.

6. Programação Orientada a Aspectos

Permite aos desenvolvedores de software separar e organizar o código de acordo com a sua importância para a aplicação, permitindo que o programador encapsule o código secundário em módulos separados do restante da aplicação.

A POA (Programação Orientada a Aspectos) permite que o programador implemente questões de segurança (log, transações bancárias...) através de aspectos, ou seja através de verificações de segurança no próprio código.

Exemplos de linguagens: existem sim linguagens que aceitam esse tipo de programação em seu código como Java, AspectJ, AspectH.

Referências

28/07/2009, http://pt.wikipedia.org/wiki/Paradigma_de_programação.

28/07/2009, http://pt.wikipedia.org/wiki/Programação_estruturada.

28/07/2009, http://pt.wikipedia.org/wiki/Programação_imperativa.

28/07/2009, http://pt.wikipedia.org/wiki/Programação_funcional.

28/07/2009, [http://pt.wikipedia.org/wiki/Pascal_\(linguagem_de_programação\)](http://pt.wikipedia.org/wiki/Pascal_(linguagem_de_programação)).