

PESQUISA PARADIGMAS
PROGRAMAÇÃO FUNCIONAL
Augusto César Araújo de Oliveira - 508991

1. Paradigma de programação imperativo ou Procedural:

Funciona mudando o estado do programa através de declarações de atribuição. O foco principal é como alcançar o objetivo, consistindo em várias declarações e, após a execução de tudo, o resultado é armazenado. As instruções devem ser passadas ao computador na sequência em que devem ser executadas. Esse tipo de programação é recomendada em projetos nos quais não se espera que haja mudanças significativas ao longo do tempo (mudanças muito frequentes), quando se tratar de uma operação complexa, ou quando não existirem muitos elementos compartilhados.

2. Paradigma declarativo:

Há declarações iniciais de verdades lógicas que são imutáveis. Ou seja, apesar de interações entre elas, o resultado será sempre igual. As principais linguagens declarativas são também as linguagens de marcação; HTML, XML, XSLT e XAML.

3. Paradigma Funcional:

Destaca o uso das funções; no momento de resolver um problema, o divide em blocos onde são implementadas as funções. As funções definem algumas variáveis que podem ou não retornar resultados. É indicado quando a resolução de algum problema depende de uma base matemática, majoritariamente.

4. Paradigma Lógico:

Utiliza formas de lógica como padrões de entrada e saída, realizando inferências para produzir os resultados. São utilizadas na solução de problemas que envolvem inteligência artificial, criação de programas especialistas e comprovação de teoremas. Exemplo: com a tentativa de provar um teorema, contendo premissas verdadeiras, a conclusão do teorema torna-se verdade também.

5. Paradigma orientado a objetos:

Mais populares do mercado atualmente. Permite a programação multiplataforma, isto é, não é necessário criar uma mesma aplicação de maneiras diferentes de acordo com o sistema operacional onde ela rodará (sites, aplicativos e softwares são desenvolvidos e interpretados por diferentes plataformas sem obstáculos). Permite a produção de programas de forma mais rápida. Linguagens como Python, Java e C++.

6. Paradigma orientado a eventos:

Definem a execução de determinadas funções conforme os eventos disparados pelo usuário. Comum no mercado de aplicações que utilizam recursos gráficos, como os jogos. Depende de uma ação prévia do usuário para efetuar um movimento. Exemplos: pressionar um botão de envio, terminar de preencher um formulário, atingir um nível específico de um jogo.