

# Programação Orientada a Objetos: Desafios e Influência da Linguagem

## PОО: Popularidade e Limitações

- A PОО, ganhando destaque na década de 1980, é agora um paradigma dominante devido à sua escalabilidade, abstração intuitiva e extensas bibliotecas.
- Ao facilitar o desenvolvimento complexo de software, a PОО não é uma solução universal.

## O Equívoco da Facilidade da PОО

- A popularidade de linguagens como C++ e Delphi decorre de um equívoco de que o domínio da PОО é facilmente alcançável, exigindo uma mudança fundamental no pensamento de desenvolvimento de software.

## A Influência da Linguagem no Pensamento e na PОО

- A linguagem molda significativamente nossa percepção da realidade, impactando a forma como abordamos a PОО.
- A linguagem influencia o pensamento, impactando a forma como percebemos e resolvemos problemas.
- Isso é visto em ambas as linguagens naturais e linguagens de programação.



## Eficiência em APL vs FORTRAN

- Uma comparação das soluções FORTRAN e APL para um problema de análise de sequência de DNA demonstra como a estrutura de uma linguagem pode levar os programadores a algoritmos diferentes e mais eficientes.
- A solução superior de programador APL ~~para um resultado~~ não da velocidade inerente da APL, mas das características da linguagem que o orientaram para uma abordagem mais eficiente.

## APL e a Criatividade na Resolução de Problemas

- A sintaxe simples da APL leva os programadores a aplicá-la ~~criativa~~ criativamente a diversos problemas, moldando sua abordagem de resolução de problemas.

## Hipótese de Sapir-Whorf e a Conjectura de Church

- A hipótese de Sapir-Whorf postula que alguns pensamentos são inexprimíveis em certas línguas.
- A conjectura de Church afirma que qualquer computação com um procedimento eficaz pode ser realizada por uma máquina de Turing.



## Linguagem e Pensamento

- Muitos linguistas também acreditam que todas as línguas são capazes de expressar qualquer ideia, embora algumas possam exigir mais esforços.
- Portanto, a linguagem influencia o pensamento, mas não limita.

## Conceitos da Programação Orientada a Objetos

- A ~~programa~~ POO é um paradigma para organizar o conhecimento e a computação.
- É semelhante ao sistema biológico de classificação de Linneu.
- O conceito central da POO envolve agentes, recebendo mensagens e cumprindo - ~~as~~ muitas vezes sem que o solicitante conheça os detalhes.
- Isso espelha a resolução de problemas do mundo real, tornando a POO intuitiva mesmo para novatos.
- Os programas orientados a objetos consistem em objetos interagentes, cada um com um papel e serviço específicos, formando uma comunidade.

## Mensagens em POO

- A programação orientada usa mensagens para iniciar ações.
- Uma mensagem enviada a um objeto, solicita uma ação.



## Método do receptor vs chamadas de procedimento

- Determina como atender à solicitação, processo que varia de acordo com o receptor.
- Isso difere das chamadas de procedimento, que não possuem um receptor designado e seleção de métodos em tempo de execução.

## Foco em responsabilidade e resultados

- O foco está nas responsabilidades ~~em~~, definindo o comportamento pelos resultados desejados, em vez de detalhes específicos da implementação.
- A programação orientada a objetos contrasta com a programação tradicional, concentrando-se no que as estruturas de dados podem fazer, não no que podem ser feito com elas.

## Herança, classes e objetos

- Objetos são instâncias de classes, herdando atributos de classes mais gerais em uma hierarquia.
- Essa herança permite uma organização e aplicação de conhecimento eficientes como visto no exemplo da compreensão de Chris sobre Fred, ~~o~~ florista.
- As classes herdam atributos das classes - mãe em uma estrutura hierárquica.
- As classes - pai abstratas, não tem instâncias diretas mas servem como modelos para subclasses.



## Substituição de métodos e hierarquia de classes

- A programação orientada a objetos usa a substituição de métodos, onde as subclasses podem substituir métodos herdados.
- A invocação do método pesquisa a hierarquia de classes, começando com a classe do receptor.

## Princípios fundamentais da POO e contraste com a programação imperativa

- Os princípios fundamentais da POO incluem objetos que se comunicam através de mensagens, cada objeto com sua própria memória e classes organizadas em uma hierarquia de herança.
- Isso contrasta com a visão tradicional da computação como um processo de manipulação de dados.
- A programação orientada contrasta com a programação imperativa tradicional, concentrando-se em objetos, mensagens e responsabilidades, em vez de endereços e variáveis de memória.

## Diferenças na implementação de Operações

- As linguagens orientadas a objetos lidam com as operações de maneira diferente.
- Algumas, como C++, usam métodos da linguagem base, enquanto outras, como Java, usam primitivas fornecidas pelo sistema.



## Origens e evolução da POO

- Os conceitos de POO originaram-se na década de 1960 com Simula.
- Alan Kay's Smalltalk, desenvolvido na década de 1970 na Xerox PARC, visava a facilidade de uso.
- Ao mesmo tempo, Bjarne Stroustrup criou C++ na Bell Labs.
- Em 1986, a POO explodiu em popularidade, com inúmeras novas linguagens surgindo.

Desde então, a programação orientada a objetos tornou-se predominante, impactando significativamente a ciência da computação.