

## **Encontro 12 - POO - Atividade**

Hélio Akira de Queiroz

PSG TECNOLOGIA APLICADA

Capacitacao PSG H3

*Luiz Augusto Rodrigues*

28/09/2021

### **Relatório referente a atividade Encontro 12 - POO - Atividade proposta em sala**

#### **Produzir uma pesquisa com o seguinte tema:**

Evolução das linguagens de programação, do Modelo Procedural ao Modelo Orientado a Objeto :

- Utilizar Normas da ABNT
- Destacar as principais linguagens de programação.
- Lista histórica e personagens.
- Destacar referências utilizadas.
- Utilizar qualquer mídia disponível (artigo, rádio, TV, vídeo, site da web).

## **Evolução das linguagens de programação, do Modelo Procedural ao Modelo Orientado a Objeto**

### **Origem**

Para que possamos entender a origem da orientação a objetos, devemos compreender os problemas que se passaram na época que foram criadas. Havia muita controvérsia acerca de assuntos que hoje já consideramos solucionados, um exemplo seria o GO TO que não era considerado má prática e hoje é abolido pela maioria dos programadores. A Orientação a objetos vem com o claro propósito de resolver os problemas que eram causados pela limitação das linguagens estruturadas e assim criar um novo paradigma.

### **Surgimento**

A Programação Orientada ao Objeto (Object-Oriented Programming) surgiu no início da década de 70, com base na linguagem SIMULA foi criada a SIMULA67 por Kristen Nygaard e Ole-Johan Dahl onde tinha seu foco em simulações realistas. Porém foi popularizada apenas com a criação da SMALLTALK pela Xerox, linguagem que foi criada puramente Orientada a Objetos.

### **Linguagens**

Para que uma linguagem possa ser considerada Orientada a Objeto, é necessário que ela consiga implementar os quatro conceitos básicos : Abstração, Encapsulamento, Herança e Polimorfismo, exemplos de linguagens que implementam esses pilares.

***C++ (1985)***

Desenvolvido por Bjarne Stroustrup com base na linguagem C e objetivo de melhorar o núcleo Unix. Porém tornou-se tão popular que é utilizada até hoje por diversas aplicações (Mozilla Firefox, Internet Explorer, Microsoft Windows, entre outros).

***Java (1991)***

Tanto um ambiente quanto uma linguagem, foi criado pela Sun Microsystems Inc, e é tido como um dos maiores representantes da Orientação a Objetos, seu diferencial era a capacidade de ser executado em qualquer arquitetura que possuísse o suporte ao interpretador.

***C#(1999)***

Baseada na linguagem C++, porém com influência de outras linguagens orientadas a objeto como Java e Object Pascal, foi criada pela Microsoft com o objetivo de ser utilizada na plataforma .NET.

**Vantagens**

Diversas são as vantagens ao empregar um paradigma orientado a objetos, em relação a comparação com a linguagem estruturada podemos citar:

***Rapidez:*** Podendo ser separado em módulos onde cada Classe pode ser trabalhada de forma individual, o que acelera a produção de sistemas grandes.

***Segurança:*** Quando criado ou alterado de forma incorreta o código, o problema não se estenderá além de sua classe, mantendo o resto do projeto sem problemas.

***Reutilizável:*** Devido a sua separação em classes, podemos reaproveitar soluções que já foram resolvidas posteriormente de forma prática, sem a necessidade de repetir as soluções dentro do código.

***Fácil Aprendizado:*** Devido a seus pilares serem baseados nas relações humanas, temos facilidade de entender seus conceitos, e podemos de forma mais rápida progredir o aprendizado.

### **Problemas**

Após a consolidação da linguagem orientada a objetos, ainda sim é muito utilizada a linguagem estruturada. Isso se deve aos problemas da orientação a objeto, que mesmo que poucos comparados com suas vantagens existem, e que podem se tornar grandes problemas dependendo do projeto aplicado. O paradigma sequencial apresenta suas vantagens quando damos atenção ao desempenho, por ser feito de forma retilínea, sua execução é mais rápida o que o torna preferencial quando o quesito é esse.

### **Resultados**

Ao analisar a fundo o paradigma orientado a objetos, compreendemos o quanto ele derivou-se do modelo estruturado, e como evoluiu a partir das novas necessidades e tecnologias que surgiram. Dados os fatos citados no artigo, podemos compreender a importância de ambos os paradigmas, tanto historicamente quanto na prática. Assim concluindo, podemos traçar um paralelo com as tecnologias atuais e assim compreender como a evolução tecnológica sempre busca a adaptação às necessidades de cada momento.

## Referências

Macoratti, José. (2014) .NET - Programação Orientada a Objetos x Programação Procedural

macoratti.net.

Acessado em 28 de setembro de 2021 pelo URL

[http://www.macoratti.net/14/06/oop\\_proc.htm](http://www.macoratti.net/14/06/oop_proc.htm)

Souza, Marcelo. (2011) Um pouco de história (POO) Techblog Desenvolvedores.

Acessado em 28 de setembro de 2021 pelo URL

<https://techblog.desenvolvedores.net/2010/12/15/um-pouco-de-historia/>

Henrique, João. (2019) POO: o que é programação orientada a objetos Alura.

Acessado em 28 de setembro de 2021 pelo URL

[https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos?gclid=CjwKCAjw-sqKBhBjEiwAVaQ9ay9W6-ykpgXBGJVNcw-CQ4X\\_ciw4Fsx-Aup1UUKLyYjeRT7wE74tnRoCfUsQAvD\\_BwE](https://www.alura.com.br/artigos/poo-programacao-orientada-a-objetos?gclid=CjwKCAjw-sqKBhBjEiwAVaQ9ay9W6-ykpgXBGJVNcw-CQ4X_ciw4Fsx-Aup1UUKLyYjeRT7wE74tnRoCfUsQAvD_BwE)

Pacievitch, Yuri. C# - C Sharp - Linguagem de Programação InfoEscola.

Acessado em 28 de setembro de 2021 pelo URL

<https://www.infoescola.com/informatica/c-sharp/>

O que é programação orientada a objetos? Apex Ensino.

Acessado em 28 de setembro de 2021 pelo URL

<https://apexensino.com.br/o-que-e-programacao-orientada-objetos/>