

Prof. Me. Renato Alves Ferreira  
email: [renato.ferreira@fmu.br](mailto:renato.ferreira@fmu.br)

Disciplina:

# Programação Orientada a Objetos

# Semana 1

## -Apresentação

- **Professor**
- **Disciplina – POO com Java**
- **Método de Avaliação**

Composição da média final =  $(N1 \cdot 0,4) + (N2 \cdot 0,6)$

- **Controle da frequência 75%** (+ - 5 dias de ausência)

## -Orientações gerais

## -Introdução a POO

**AVALIAÇÃO**

A Média Final (MF) da disciplina considera os seguintes elementos e valores:

N1	N2
<p><b>A1 – Avaliação(ões)</b> a ser(em) definida(s) de acordo com os objetivos de aprendizagem</p> <p><b>[nota de 0 a 10]</b></p>	<p><b>A2 – Avaliação(ões)</b> a ser(em) definida(s) de acordo com os objetivos de aprendizagem (9,0 pontos) + <b>APS –</b> Atividade Prática Supervisionada (1,0 ponto)</p> <p><b>[nota de 0 a 10]</b></p> <p><b>OU</b></p> <p><b>SUB – Avaliação Substitutiva</b></p> <p><b>[nota de 0 a 10]</b></p>

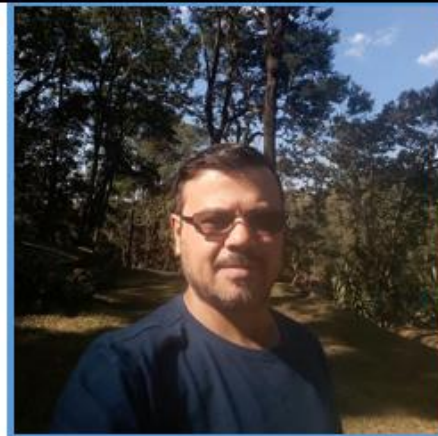
A Média Final (MF) é calculada por meio da média ponderada das duas notas, N1 e N2, com peso, respectivamente de 40% e 60%, resultante da seguinte equação:

$$MF = (N1*0,4) + (N2*0,6)$$

Para aprovação, a Média Final deverá ser igual ou superior a 6,0 (seis), além da necessária frequência mínima de 75% nas aulas.

O estudante que não realizar a A2 ou não atingir a média final 6,0 (seis) na disciplina, poderá realizar uma Avaliação Substitutiva (SUB), cuja nota substituirá a nota de A2 obtida, caso seja maior.

## Breve currículo do professor



Mestre em Informática e Gestão do Conhecimento, Especialista em Sistema de Informação com ênfase em TI, Téc. em Processamento de Dados, Pedagogo, professor de cursos médio e técnico desde 1989 e no curso superior, desde 2004. Escritor e Contadante de vários títulos na área da TI. Cursos em infraestrutura, como CCNA/CISCO e MS Windows Server, cursos de Robótica/Automação/IoT, inclusive no exterior (University of Applied Sciences Upper Austria). Concomitantemente à carreira acadêmica, na área corporativa atuação como desenvolvedor de soluções de softwares e redes de 1989 a 2013 e microempresário de 1996 a 2005.



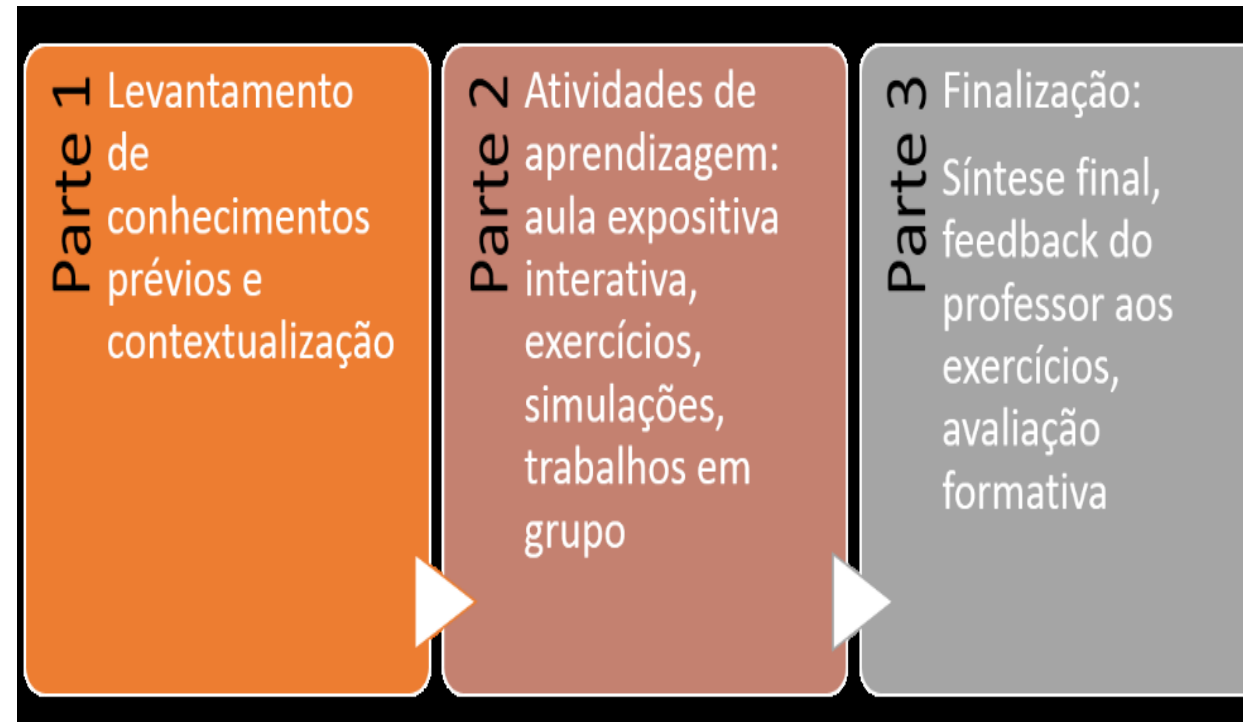
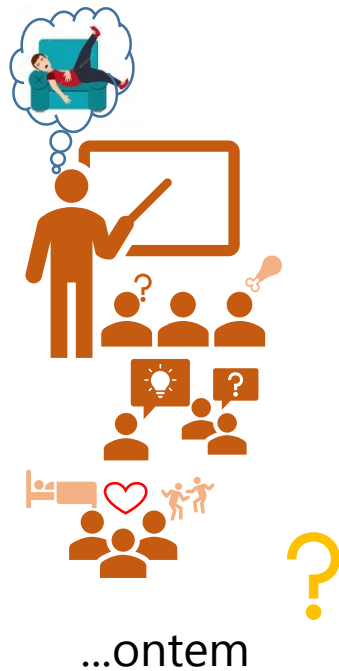
- Email : [renato.ferreira@fmu.br](mailto:renato.ferreira@fmu.br)
- LinkedIn : <https://www.linkedin.com/in/renato-alves-ferreira/>
- Lattes : <http://lattes.cnpq.br/4524836424246846>

## ORIENTAÇÕES ACADÊMICAS (portfólio do estudante)

“Para tornar-se um profissional competente naquilo que faz, o mercado de trabalho exige que você mantenha uma atitude de buscar aprender sempre, de modo cada vez mais ativo e autônomo.”

Pensando nisso, suas aulas utilizam **metodologias ativas**, que buscam levá-lo(a) a envolver-se nas atividades e fomentar uma aprendizagem realmente significativa.

As aulas são estruturadas em 3 partes:



# Postura universitária em ambiente acadêmico

“Uma vida acadêmica bem conduzida permite que você compreenda e se aproprie de conhecimentos.”

No entanto, isso não acontecerá de forma passiva: por melhor que seja o plano de aula do professor ou sua didática, só VOCÊ pode REALMENTE construir seus conhecimentos. Por isso, é essencial que você mantenha uma atitude positiva de aprendizagem, que se traduz em:

- Chegar no horário de início e ficar até o final da aula;
- Participar ativamente das propostas de trabalho de cada aula;
- Anotar as explicações e orientações do professor(a);
- Respeitar as opiniões divergentes de colegas ou do professor;
- Buscar fundamentar suas opiniões com dados científicos;
- Fazer os exercícios indicados como atividades extraclasse.

(portfólio do estudante)

# Seja Resiliente

A **resiliência** é a capacidade do indivíduo lidar com problemas, adaptar-se a mudanças, superar obstáculos ou resistir à pressão de situações adversas - choque, estresse, algum tipo de evento traumático, sem entrar em surto psicológico, emocional ou físico, por encontrar soluções estratégicas para enfrentar e superar as adversidades.

Nas organizações, **resiliência** trata-se de uma tomada de decisão ao se deparar com um contexto entre a tensão do ambiente e a vontade de vencer. Essas decisões propiciam forças estratégicas para enfrentar a adversidade.

(Wikipédia)



VOCÊ EM AÇÃO !



# COMO APRENDEMOS



A pirâmide de aprendizagem  
de Willian Glasser





# O que esperar da disciplina

- Abordagem dos principais fundamentos da POO;
- Modelar e implementar, problemas de pequena e média complexidade em Java;
- Abstrair o contexto do problema no mundo real para a orientação a objetos;
- Desenvolver programas levando em consideração conceitos da OO, como:
  - Reuso de código;
  - Encapsulamento;
  - Herança;
  - Polimorfismo;
  - Abstração;
  - etc.

# REFERÊNCIAS

## BIBLIOGRAFIA BÁSICA

- ✓ FURGERI, S. Programação orientada a objetos: Conceitos e Técnicas. São Paulo: Érica, 2016. 168p.
- ✓ MANZANO, J. A. G.; COSTA JR., R. Programação de Computadores com Java. Érica, 2014. 127p. [Minha Biblioteca]
- ✓ MANZANO, J. A. G. Programação de Computadores com C/C++. Érica, 06/2014. 120p. [Minha Biblioteca].

## BIBLIOGRAFIA COMPLEMENTAR

- ✓ BARNES, D. J.; KOLLING, M. Programação Orientada a Objetos com Java: uma introdução prática usando o BlueJ - 4ª edição. São Paulo: Pearson Prentice Hall, 2009. 480p. [Biblioteca Virtual Universitária].
- ✓ MEILIR, P. Fundamentos do Desenho Orientado a Objeto com UML. São Paulo: Makron Books, 2001. 462p. [Biblioteca Virtual Universitária].
- ✓ FÉLIX, R. Programação Orientada a Objetos. São Paulo: Pearson Education do Brasil, 2016. 164p. [Biblioteca Virtual Universitária].
- ✓ KOFFMAN, E. B., WOLFGANG, P. T. Objetos, Abstração, Estrutura de Dados e Projeto Usando C++. Rio de Janeiro: LTC, 2008. 455p. [Minha Biblioteca].
- ✓ FURGERI, S. Java 8 - Ensino Didático - Desenvolvimento e Implementação de Aplicações. São Paulo: Érica, 2015. 320p. [Minha Biblioteca].

# Introdução a POO - Conceitos iniciais

- **Paradigmas das linguagens de programação**
- **Relembrar o paradigma da Prog. Estruturada**
- **Conceituar a Programação Orientada a Objetos**
- **Vantagens e desvantagens entre paradigmas**
- **Considerações finais**

# Paradigmas das linguagens de programação

Alguns exemplos:

- Imperativa/Procedural – procedimentos e rotinas : Basic, Ada, Fortran, Cobol, Assembly
- Estruturada – blocos estruturados que controlam o fluxo do programa; uso funções e procedures: Pascal, C, Clipper
- Orientada a Objetos – maior abstração, organização e padronização com a manipulação de objetos : Java, Python, C#, C++ , Smalltalk, Simula

# Programação Estruturada X Orientada a Objetos

## Estruturada

Vantagem : Controle mais eficaz quanto ao fluxo de execução do programa e a facilidade em compreender o código como um todo.

Desvantagem : Facilidade em desenvolver códigos confusos e reuso de código menos eficiente.

## Orientada a Objetos

Vantagem : Maior produtividade com o reuso de código e divisão de tarefas em projetos de desenvolvimento, melhor organização geral da codificação. Maior padronização.

Desvantagem : Desempenho do código normalmente é inferior aos outros paradigmas. Aplicar e perceber alguns dos conceitos de orientação a objetos não é uma tarefa trivial.

# Comparativo entre os paradigmas

<b>Programação Orientada a Objetos</b>	<b>Programação Estruturada</b>
Classes	Tipos de dados definidos pelo usuário
Instâncias de variáveis	Variáveis
Métodos	Procedimentos e funções
Mensagens	Chamadas a procedimentos e funções
Herança	Não disponível
Polimorfismo	Não disponível
Objetos	Não disponível
Reuso de código facilitado	Maior dificuldade e limitação
Divisão das tarefas mais eficiente	Limitada
Maior organização geral	Menor organização geral



# Apresentação dos principais elementos da POO

- Classe
- Objeto
- Instância
- Atributo
- Método
- Herança
- Encapsulamento
- Abstração
- Polimorfismo

(serão discutidos e exemplificados no decorrer das aulas)



Durante o curso:

- Leitura do artigo indicado (Recurso 1)
- Leitura do livro indicado (Recurso 2)

### **Recurso 1**

Artigo Devmedia: “Principais conceitos da Programação Orientada a Objetos” Disponível em:

<https://www.devmedia.com.br/principais-conceitos-da-programacao-orientada-a-objetos/32285>

### **Recurso 2**

Livro: FÉLIX, R. Programação Orientada a Objetos. São Paulo: Pearson Education do Brasil, 2016. 164p.  
[Biblioteca Virtual Universitária] pp 1-4.



# Te espero na próxima aula!