

# Programação e Desenvolvimento de Software 2

## Apresentação do curso

Profs. Luiz Chaimowicz

(slides adaptados do Prof. Douglas Macharet)



# Objetivos do curso

Apresentar técnicas básicas de desenvolvimento, teste e análise de programas de computador, para a resolução de problemas de forma eficaz. É esperado que nesta disciplina os alunos desenvolvam seus primeiros programas de tamanho moderado, motivando a necessidade de uso de boas práticas de desenvolvimento, fixando os conteúdos abordados através de atividades práticas. Concluindo o curso, os alunos deverão dominar as técnicas mais básicas utilizadas no processo de desenvolvimento de software, incluindo os conceitos básicos de Orientação a Objetos.

# Ementa

## FUNDAMENTOS

- Memória, ponteiros, alocação dinâmica, TADs

## BOAS PRÁTICAS

- Formatação, comentários, controle de versão

## PROGRAMAÇÃO ORIENTADA A OBJETOS

- Abstração, encapsulamento, herança, polimorfismo

## CORRETUDE

- Depuração, tratamento de exceções, testes

# Bibliografia

- **Clean Code: A Handbook of Agile Software Craftsmanship.**  
Robert C. Martin.  
Prentice Hall, 2008.
- **Code Complete: A Practical Handbook of Software Construction.**  
Steve McConnell.  
Microsoft Press, 2004. 2nd Edition.
- **Effective C++: 55 Specific Ways to Improve Your Programs and Designs.**  
Scott Meyers.  
Addison-Wesley Professional, 2005. 3rd Edition.
- **A Tour of C++.**  
Bjarne Stroustrup.  
Addison-Wesley Professional, 2013. 1st Edition.
- outros....

# Critérios de avaliação

- **2 Provas (2x20): 40 pontos**
- **Atividades práticas: (VPL): 40 pontos**  
(~1 Exercício por semana)
- **Projeto final: 20 pontos**

# Critérios de avaliação

## ■ Provas

- Conteúdo visto durante a aula
- Material de referência (slides, livros, etc.)
- Exercícios de laboratório

## ■ Em caso de falta

- Justificativa em até 48 horas após data de realização
- Outra avaliação será dada ao final (todo conteúdo!)

# Critérios de avaliação

- Laboratórios Virtuais de Programação (VPL)
  - Ambiente de programação integrado ao Moodle
  - O aluno foca exclusivamente em resolver o exercício
  - Correção automática utilizando diferentes casos de teste
  - **Eventuais entrevistas para complementar a avaliação**
- Múltiplas tentativas → Maior nota (última submissão)
  - Tente fazer localmente para não sobrecarregar o Sistema
  - **Lembre-se sempre de apertar o botão avaliar!**

# Critérios de avaliação

- Projeto Final (em grupo)
  - Código
    - Funcionamento, aplicação dos conceitos, ...
  - Documentação
    - Clareza e coesão, conteúdo, ...
  - Controle de Versão, Testes
  - Pontos extras: criatividade, recursos adicionais, etc.
- Tema será divulgado em breve...



# Honestidade acadêmica

- Haverá tolerância **ZERO** com cópia/cola
  - Nota da atividade será automaticamente anulada
  - Poderá ser aberto processo disciplinar no colegiado



Compartilhe dicas e sugestões,  
mas não compartilhe código!

# Honestidade acadêmica

- Uso de IA Generativa (ChatGPT, etc)
  - “Se você deixar a IA te substituir hoje, provavelmente você também será substituído no futuro.”
  - Evite usar, mas se usar, **entenda** o que ela fez e **seja crítico** com os resultados



Use com moderação!

# Moodle

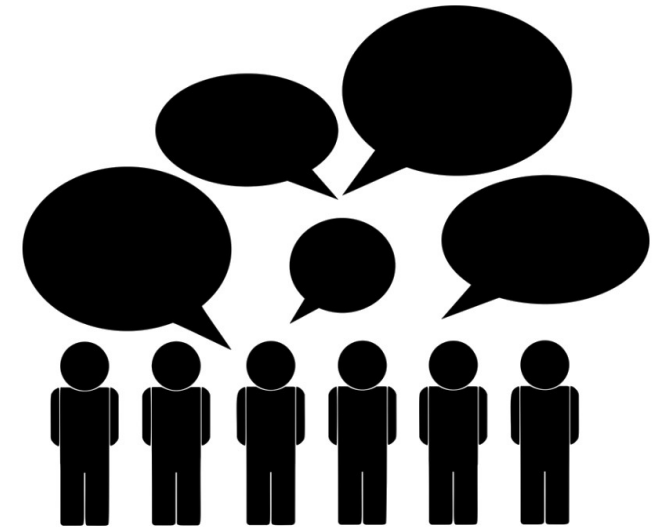
- **Todas** as informações do curso
  - Avisos
  - Notas de aulas
  - Discussão de dúvidas
  - Atividades práticas
  - Projeto
- Metaturma compartilhada com outras turmas (Prof. Héctor BSI)



# Participação

- Lista de presença em todas as aulas
- Por que devo participar?
  - Tirar dúvidas (suas e de colegas)
  - Colaborar com discussões em sala
  - Dicas sobre possíveis questões
  - Mencionar algo não contido nos slides

Lembre-se, são roteiros de aula e não uma apostila!



# Monitoria

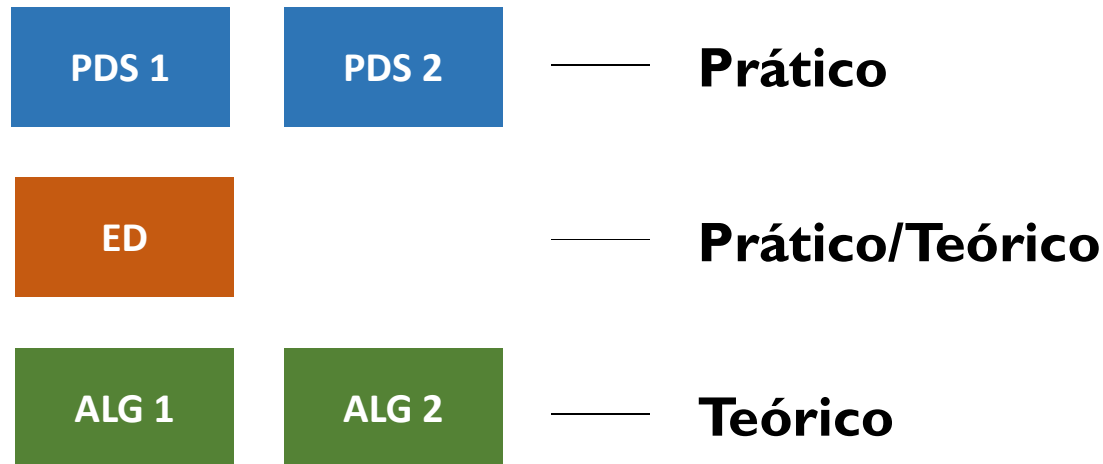
- Horários semanais (a definir)
- Faça uso constante do **Fórum**
  - Maior interação entre a própria turma
  - Vários alunos podem ter a mesma dúvida
  - Outras questões não abordadas diretamente
- Não tenha medo/vergonha da sua dúvida!

# Contato

- Email: casos especiais ou de cunho particular
  - [chaimo@dcc.ufmg.br](mailto:chaimo@dcc.ufmg.br)
  - Adicionar no assunto: **[DCC204]**
- Sala:
  - ICEx – 6328 (Anexo U)
  - Agendar um horário por email

# PDS 2

- O que você ouviu sobre PDS 2?
- Quais as suas expectativas para o curso?

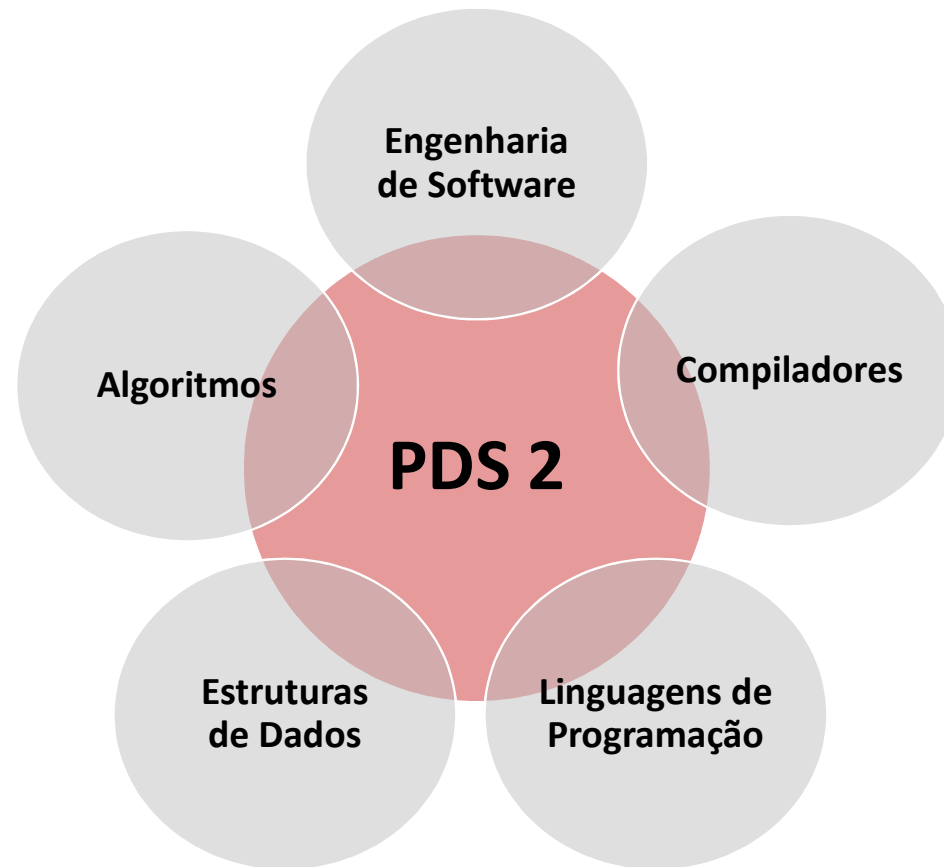


# PDS 2

- Não é um curso técnico de programação!
  - Linguagem é uma ferramenta para aplicar os conceitos
  - O foco não é pura e simplesmente na sintaxe!
- Objetivos
  - Desenvolvimento de **bons programas** (e profissionais)
    - Ganhar experiência prática (extraclasse) em programação
    - Grandes projetos → Grandes responsabilidades
  - Foco nos princípios e conceitos do paradigma OO



# PDS 2



# PDS1 vs. PDS2

## ■ PDS1

- Introdução básica
  - variáveis, if, else, for, while, funções, ...
- Maior acompanhamento / Lento



## ■ PDS2

- Conteúdo teórico extenso/complexo
  - POO, boas práticas, ferramentas, ...
- Mais autonomia do aluno / Rápido



# Active Learning – Depende mais de você!

- Conteúdo é **CUMULATIVO**
- Exige dedicação e prática!
- Principal motivo de reprovação
  - Acúmulo → Desistência
  - Dedicação → Aprovação



# Tarefas

- Acessar o Moodle, ver material disponível
- Preparar ambiente de desenvolvimento<sup>1</sup>
- Fazer os exercícios de ambientação VPL

<sup>1</sup> <https://www.youtube.com/watch?v=8motuhVylEg>  
<https://www.youtube.com/playlist?list=PLgrZRt8Cq-J7qXyWPzDsVUaKTc4gHZ-1L>

<sup>2</sup> [https://www.youtube.com/watch?v=yD\\_GIJDtSo](https://www.youtube.com/watch?v=yD_GIJDtSo)