

Programação Orientada a Objetos

Aula 00 – Apresentação

Hugo Marcondes

Departamento Acadêmico de Eletrônica
DAELN

hugo.marcondes@ifsc.edu.br



INSTITUTO
FEDERAL
Santa Catarina
Câmpus
Florianópolis

Introdução

Objetivo

Disponibilizar ao aluno conhecimentos acerca dos conceitos de programação orientada a objetos, de forma que o aluno adquira competência para desenvolver e implementar aplicações segundo este paradigma de programação de forma eficiente.

- Carga horária: 80ha
 - Segunda-feira – 18:30 às 22:30
- Atendimento Extra-Classe
 - Segunda-feira – 16:30 às 17:30
 - Quarta-feira – 08:30 às 09:30
 - Agendar com antecedência (E-mail)
 - Presencial / Google Meet



Competências, Habilidades e Atitudes

- Ao término da unidade curricular, o estudante deve ser capaz de **compreender** as etapas necessárias para o desenvolvimento de programas utilizando o **paradigma de orientação a objetos**. Dentre as habilidades esperadas do aluno, tem-se:

- Especificar e modelar sistemas utilizando a linguagem UML;
- Implementar o sistema especificado utilizando uma linguagem de programação orientada a objetos;
- Utilizar uma biblioteca gráfica para construir aplicações com interfaces gráficas;



Atitudes

- Criatividade e iniciativa do estudante
- Interação em trabalhos de grupo
- Manifestações de interesse
- Organização
- Assiduidade
- Atitudes poderão influenciar o seu conceito final!

Notes

Notes

Notes

Notes



Conteúdo

- Análise e Projeto de Sistemas Orientados à Objetos
 - Introdução: Paradigmas de Programação
 - Conceitos de Orientação à Objetos
 - Classes; Relações e Polimorfismo
 - Modelos UML
- Linguagens de Programação Orientada à Objetos
 - Linguagem C++
 - Biblioteca Padrão C++: STL
 - Sobrecarga de operadores
 - Metaprogramação Estática
- Framework de Desenvolvimento QT
- Desenvolvimento de Projeto Prático



Estratégias de Ensino

- Aulas expositivas **dialogadas**
- Resolução de exercícios
- **Atividades práticas**



Avaliação

$$CF = 0,25 * AV1 + 0,25 * AV2 + 0,25 * AV3 + 0,25 * AT$$

- Trabalho prático
 - Uma avaliação teórica (AV1)
 - Uma avaliação prática (AV2)
 - Duas avaliações do projeto final (AV3)
- CF – Conceito final
- AV1, AV2, AV3 – Avaliações
- AT – Participação nas atividades presenciais
 - Presença / participação nas aulas
 - Entregas das atividades propostas



Projeto prático

- Desenvolvimento de uma aplicação orientada à objetos
 - Tema de escolha livre
 - Times de desenvolvimento
 - Data limite para definição do tema
- Uso obrigatório do GitHub para o desenvolvimento do projeto
 - Organização é fundamental
 - Frequência de commits (por todos integrantes)
 - Utilização correta do GitHub (controle de versões)
 - Utilizado também para a documentação do código
 - Markdown pages (README.md)
 - GitHub pages
 - Wiki
 - Project

Notes

Notes

Notes

Notes



Avaliação

- UML – Elaboração e corretude dos diagramas (25%)
- GITHUB – Organização do repositório, frequência de commits, uso correto do GITHUB (10%)
- Documentação – Clareza na apresentação e organização final do projeto (10%)
- Implementação – Correção do código, uso correto da linguagem, uso correto da biblioteca (55%)



Informações Gerais

- A reposição de atividades só é permitida com apresentação de atestado médico (no caso das avaliações) e justificativa apropriada, conforme define o regimento didático-pedagógico da instituição. Se deferida pela Coordenação, será realizada em horário a ser marcado com o docente da disciplina.
- Os equipamentos de laboratório devem ser mantidos organizados.
- Material da disciplina será disponibilizado através do SIGAA e do repositório da disciplina no GITHUB.



Bibliografia

Bibliografia Adicional



That's all folks!



Notes

Notes

Notes

Notes
