
UNIVERSIDADE FEDERAL DE MINAS GERAIS

Escola de Engenharia

Curso de Graduação em Engenharia de Controle e Automação

PROGRAMA DE DISCIPLINA**DISCIPLINA:**

Análise, Projeto e Programação Orientada a Objetos

CÓDIGO:

ENG075-001

DEPARTAMENTO RESPONSÁVEL:

Departamento de Engenharia

UNIDADE:

Escola de Engenharia

CARGA HORÁRIA:

Teórica: 60

Prática:

CRÉDITOS:

4

PERÍODO:

a partir do 5º

CLASSIFICAÇÃO:

OP

PRÉ-REQUISITOS:

Total: 60 horas-aula

PRÉ-REQUISITOS DE CONHECIMENTO:

Programação de computadores, preferencialmente em linguagem C

CARGA HORÁRIA ESTIMADA EXTRA-CLASSE:

120 h

EMENTA

Gerenciamento da Complexidade; Modelagem Orientada a Objetos; Classes, Objetos, Métodos, Encapsulamento, Herança e Composição; Diagramas e Implementação; Polimorfismo; Tratamento de exceções; Parametrização de classes.

OBJETIVOS:

O aluno, ao final do semestre letivo, deve ser capaz de compreender os princípios da análise, projeto e programação orientados a objetos, de maneira que, dado um problema de engenharia, ele seja capaz de modelar, projetar e implementar uma solução utilizando a tecnologia da orientação a objetos. A disciplina apresenta, em aulas teóricas, os conceitos fundamentais e estes são utilizados pelos alunos para a implementação prática de soluções de Engenharia em seus trabalhos práticos

METODOLOGIA DE ENSINO:

Aulas Expositivas em Quadro-Negro
 Utilização de Transparências ou Slides
 Aulas Práticas Demonstrativas
 Aulas Práticas de Montagem
 Trabalho Teórico Extra-Classe

Trabalho Prático Extra-Classe
 Estudo Dirigido / Listas de Exercícios
 Aulas em Salas de Microcomputadores
 Outros - Especificar:

CRITÉRIOS DE AVALIAÇÃO:

2 provas teóricas – 60 pontos

4 trabalhos práticos – 40 pontos

PROGRAMA:

Unidade I: Gerenciando a Complexidade

- I.1. A complexidade inerente dos sistemas de software
- I.2. A estrutura de sistemas complexos
- I.3. Trazendo ordem ao caos
 - I.3.1. Decomposição
 - I.3.2. Abstração
 - I.3.3. Hierarquia
- I.4. Sobre o projeto de sistemas complexos

Unidade II: Modelagem orientada a objetos

- II.1. A Evolução dos modelos orientados a objetos
- II.2. A abstração de objetos: encapsulamento, interface e implementação .
- II.3. Reutilizando a implementação
- II.4. Reutilizando a interface através de herança
- II.5. Objetos intercambiáveis: polimorfismo
- II.6. Introdução à Análise e ao Projeto Orientado a Objetos utilizando a UML (Unified Modeling Language)

Unidade III: Classes e Objetos:

- III.1. Implementando classes e objetos em C++
- III.2. Atributos e métodos: controle de acesso e encapsulamento
- III.3. Sobrecarga de funções e argumentos default
- III.4. Inicialização e destruição
- III.5. Constantes , funções inline e controle de visibilidade
- III.6. Ponteiros, referências, atributos dinâmicos, gerenciamento de memória e o construtor de cópia
- III.7. Sobrecarga de operadores e conversão de tipos

Unidade IV: Herança e Composição

- IV.1. Herança simples
- IV.2. Composição
- IV.3. Combinando composição e herança
- IV.4. Herança múltipla
- IV.5. Polimorfismo e funções virtuais
- IV.6. Classes abstratas

Unidade V: Outros Mecanismos

- V.1. Classes parametrizadas
- V.2. A Biblioteca padrão
- V.3. Exceções
- V.4. Identificação de tipo em tempo de execução

Unidade VI: Revisitando a Análise e o Projeto Orientados a Objetos

- VI.1. Modelagem Estrutural
- VI.2. Modelagem comportamental
- VI.3. Modelagem de Arquitetura
- VI.4. Padrões de projeto

BIBLIOGRAFIA:

1. Bruce Eckel, *Thinking in C++ - vols 1 & 2*, 2nd Edition, Prentice Hall, 1999 (também está disponível a [versão impressa do volume 1](#) do livro)
2. Harvey M. Deitel & Paul J. Deitel, *C++ como programar*, Bookman Companhia Ed., 2001 (existe a versão em inglês do livro, [também na forma multimídia](#)).
3. Grady Booch, Ivar Jacobson e James Rumbaugh. *The Unified Modeling Language User Guide*. Addison-Wesley, Reading - MA, 1999.
4. [UML Multimedia Cyber Classroom](#), Addison Wesley Longman, Pearson PTR. (OBS: [UML CyberClassRoom False Positive for Virus](#))
5. Bjarne Stroustrup *The C++ Programming Language*, third edition, Addison-Wesley, 1997 (Também existe uma versão mais nova (e mais cara) deste livro: [The C++ Programming Language, Special Edition](#))
6. Terry Quatrani. *Visual Modeling with Rational Rose and UML*. Addison-Wesley, Reading - MA, 1998. (Existe também a versão 2000 deste livro).
7. S. B. Lippman & J. Lajoie, *C++ Primer*, third edition, Addison-Wesley, 1998.
8. James Rumbaugh, Ivar Jacobson e Grady Booch. *Unified Modeling Language Reference Manual*. Addison-Wesley, Reading - MA, 1999.
9. Grady Booch. *Object-Oriented Analysis and Design with Applications* 2nd. ed. Benjamin/ Cummings, Redwood City - CA, 1994.

UNIVERSIDADE FEDERAL DE MINAS GERAIS

Escola de Engenharia

Curso de Graduação em Engenharia de Controle e Automação

PROFESSOR RESPONSÁVEL:

Renato Cardoso Mesquita

DATA DA APROVAÇÃO:

--