

DISCIPLINA: Programação de Computadores II	CÓDIGO: 2ECOM.007
---	--------------------------

VALIDADE: a partir de agosto de 2010

TÉRMINO:

Carga Horária: Total: 30 horas-aula Semanal: 02 aulas Créditos: 02

Modalidade: Teórica

Classificação do Conteúdo pelas DCN: Básica

Ementa:

Conceitos de orientação a objetos: tipos abstratos de dados, objetos, classes, métodos, visibilidade, escopo, encapsulamento, associações de classes, estruturas todo-parte e generalização-especialização, interfaces; herança de interface e de classe, polimorfismo, sobrecarga, invocação de métodos; aplicações em uma linguagem de programação orientada a objetos; noções de modelagem de sistemas usando UML: diagrama de classes e de interação.

Curso (s)	Período	Eixo	Natureza
Engenharia de Computação	2	Fundamentos de Engenharia de Computação	Obrigatória
Engenharia Elétrica	2	Computação e Matemática Aplicada	Obrigatória
Engenharia Mecânica	2	Computação e Matemática Aplicada	Obrigatória
Engenharia de Produção Civil	3	Computação e Matemática Aplicada	Obrigatória
Engenharia Mecatrônica	2	Programação de Computadores e Computação Aplicada	Obrigatória
Engenharia de Controle e Automação	2	Computação e Matemática Aplicada	Obrigatória
Engenharia de Materiais	3	Linguagem de Programação e Expressão Gráfica	Obrigatória
Engenharia de Automação Industrial	2	Computação e Matemática Aplicada	Obrigatória

Departamento/Coordenação: Departamento de Computação (DECOM)

INTERDISCIPLINARIEDADES

Pré-requisitos
- Programação de Computadores I. - Laboratório de Programação de Computadores I.
Co-requisitos
- Laboratório de Programação de Computadores II.
Disciplinas para as quais é pré-requisito
- Algoritmos e Estruturas de Dados I (Engenharia de Computação) - Linguagens de Programação (Eng. Mecatrônica) - Análise e Projeto de Algoritmos (Eng. Mecatrônica) - Tópicos Especiais em Programação de Computadores e Computação Aplicada (Eng. Mecatrônica) - Métodos Numéricos Computacionais (Eng. de Computação; Eng. Elétrica, Eng. Mecânica, Eng. de Produção Civil, Eng. Mecatrônica, Eng. de Controle e Automação, Eng. de Materiais e Eng. de Automação Industrial)
Disciplinas para as quais é co-requisito
- Lab. de Programação de Computadores II
Transdisciplinariedade (inter-relações desejáveis)

Objetivos: <i>A disciplina deverá possibilitar ao estudante</i>
- Conhecer e saber utilizar os conceitos de programação orientada a objetos. - Projetar e implementar programas utilizando o paradigma de orientação a objetos.

Unidade de ensino		Carga-horária Horas-aula
1	Introdução a C++ e aos fundamentos de orientação a objetos <ul style="list-style-type: none">Contexto histórico das linguagens de programaçãoParadigmas de programaçãoFundamentos das linguagens orientadas a objetos	2
2	Programação em C++ <ul style="list-style-type: none">Conceitos básicos em C++Constantes, tipos, operadoresEstruturas de controleArrays e apontadoresAlocação dinâmica de memória	4
3	Funções, sobrecarga e namespaces <ul style="list-style-type: none">Namespaces e escopoFunções: declaração e definiçãoFunções recursivasFunções inlineSobrecarga de funçõesPassagem de parâmetro por cópia e por referência	4
4	Introdução a classes e objetos <ul style="list-style-type: none">AbstraçãoEncapsulamentoTipos abstratos de dadosClasses: definição e usoObjetos: instâncias de classes	2
5	Funções implícitas e sobrecarga de operadores <ul style="list-style-type: none">Funções construtoras e destruidorasSobrecarga de operadoresApontador this	2
6	Tipos específicos de funções e amizade <ul style="list-style-type: none">Funções constantesFunções estáticasFunções amigasClasses amigas	2
7	Herança <ul style="list-style-type: none">Visibilidade, escopo, especificadores de acessoClasses base e derivadaHerança de construtores e destruidoresHierarquia de classesHerança múltipla	6
8	Polimorfismo de tipos <ul style="list-style-type: none">Funções virtuais e virtuais puras	

Plano de Ensino

	<ul style="list-style-type: none">• Classes abstratas• Tipos polimórficos• Uso do polimorfismo	4
9	Outros tópicos <ul style="list-style-type: none">• Formatação de saída• Templates ou gabaritos• Tratamento de exceção	2
10	Noções de modelagem com UML <ul style="list-style-type: none">• Diagrama de classes• Diagrama de interação	2
Total		30

Bibliografia Básica

LAFORE, R. **Object-oriented programming in C++**. 4th ed. Indianapolis: Sams, 2002.

Bibliografia Complementar

DEITEL, H. M.; DEITEL, P., J. **Java: como programar**. 8. ed. São Paulo: Pearson Prentice-Hall, 2010.

LIPMAN, S. B.; LAJOIE, J. **A C++ primer**. 4th ed. New York: Addison-Wesley, 2005.