

DISCIPLINA: Lab. de Programação de Computadores II | **CÓDIGO:** 2ECOM.008

VALIDADE: a partir de agosto de 2010

TÉRMINO:

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

Modalidade: Prática

Classificação do Conteúdo pelas DCN: Básica

Ementa:

Práticas em laboratório dos temas e tópicos abordados na disciplina "Programação de Computadores II".

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
Co-requisitos
- Programação de Computadores II.
Disciplinas para as quais é pré-requisito
- Lab. de Algoritmos e Estruturas de Dados I (Engenharia de Computação)
Disciplinas para as quais é co-requisito
- 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 programação• Paradigmas de programação• Fundamentos das linguagens orientadas a objetos	2
2	Programação em C++ <ul style="list-style-type: none">• Conceitos básicos em C++• Constantes, tipos, operadores• Estruturas de controle• Arrays e apontadores• Alocação dinâmica de memória	4
3	Funções, sobrecarga e namespaces <ul style="list-style-type: none">• Namespaces e escopo• Funções: declaração e definição• Funções recursivas• Funções inline• Sobrecarga de funções• Passagem de parâmetro por cópia e por referência	4
4	Introdução a classes e objetos <ul style="list-style-type: none">• Abstração• Encapsulamento• Tipos abstratos de dados• Classes: definição e uso• Objetos: instâncias de classes	2
5	Funções implícitas e sobrecarga de operadores <ul style="list-style-type: none">• Funções construtoras e destruidoras• Sobrecarga de operadores• Apontador this	2
6	Tipos específicos de funções e amizade <ul style="list-style-type: none">• Funções constantes• Funções estáticas• Funções amigas• Classes amigas	2
7	Herança <ul style="list-style-type: none">• Visibilidade, escopo, especificadores de acesso• Classes base e derivada• Herança de construtores e destruidores• Hierarquia de classes• Herança múltipla	6

Plano de Ensino

8	Polimorfismo de tipos <ul style="list-style-type: none">• Funções virtuais e virtuais puras• 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**. 6. ed. São Paulo: Prentice-Hall, 2005.

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