

UNIDADE:			
<input type="checkbox"/> DES	<input type="checkbox"/> Campus - Leopoldina	<input checked="" type="checkbox"/> Campus - Araxá	<input type="checkbox"/> Campus - Divinópolis
CURSO: ENGENHARIA DE AUTOMAÇÃO INDUSTRIAL			

DISCIPLINA Programação Orientada a Objetos	<input type="checkbox"/> TEÓRICA <input type="checkbox"/> PRÁTICA <input checked="" type="checkbox"/> TEORICA PRATICA	VÁLIDO A PARTIR DE SEMESTRE <input checked="" type="checkbox"/> 1º <input type="checkbox"/> 2º ANO 2016
DEPARTAMENTO/ COORDENAÇÃO: Coordenação de Eng. de Automação Industrial		CÓDIGO: OPT XI
CARGA HORÁRIA TOTAL 36 Horas-aula	CARGA HORÁRIA SEMANAL 2 Horas-aula	CRÉDITOS: 2
		PERÍODO(S): 9º e 11º
NÃO TEM <input type="checkbox"/> PRÉ-REQUISITOS <input checked="" type="checkbox"/> CO-REQUISITOS		
REQUISITOS	PRÉ-REQUISITOS	
	DISCIPLINA(S)	CÓDIGO(S)
	PROGRAMAÇÃO COMPUTACIONAL 2	PRG02
CO-REQUISITOS		
DISCIPLINA(S)		CÓDIGO(S)
EMENTA	ANÁLISE E PROJETO ORIENTADO A OBJETOS. NOTAÇÃO UML (UNIFIED MODELING LANGUAGE). LINGUAGEM DE PROGRAMAÇÃO ORIENTADA A OBJETOS. TIPOS ABSTRATOS DE DADOS. CLASSES. OPERAÇÕES (MÉTODOS). ATRIBUTOS. HERANÇA. POLIMORFISMO. PROGRAMAÇÃO GENÉRICA (TEMPLATES). PADRÕES DE PROJETO.	

OBJETIVOS
A disciplina deverá possibilitar ao estudante: <ul style="list-style-type: none"> • Conceber e implementar soluções de software utilizando o paradigma orientado a objetos; • Interpretar e criticar modelos de software representados em UML; • Aplicar metodologias de desenvolvimento de projetos que contemplem as fases de análise de requisitos, definição e refinamento da arquitetura, implementação, testes e validação; • Usar adequadamente abstração de dados, herança e padrões de projeto para reutilização de código.

TÉCNICAS UTILIZADAS (Marque com um X no quadro):	
<input type="checkbox"/> Aula expositiva em quadro <input type="checkbox"/> Aula com uso de transparência <input checked="" type="checkbox"/> Aula com uso de multimídia <input checked="" type="checkbox"/> Aula prática <input type="checkbox"/> Discussão de texto <input type="checkbox"/> Filme	<input type="checkbox"/> Seminário <input type="checkbox"/> Pesquisa <input checked="" type="checkbox"/> Trabalho individual <input checked="" type="checkbox"/> Trabalho em grupo <input type="checkbox"/> Visita técnica <input type="checkbox"/> Outros

DISCIPLINA	Programação Orientada a Objetos
-------------------	---------------------------------

UNIDADES DE ENSINO	Carga Horária (horas-aula)
UNIDADE 1 – Fundamentos da Programação Orientação a Objetos 1.1 – Paradigmas de programação 1.2 – Tipos abstratos dados, classes e objetos 1.3 – Operações, atributos e propriedades 1.4 – Encapsulamento 1.5 – Herança e Polimorfismo 1.6 – Metodologias de Desenvolvimento	4 h-a
UNIDADE 2 – Introdução à Notação UML (Unified Modeling Language) 2.1 – Diagramas de Casos de Uso 2.2 – Diagramas de Classes 2.3 – Diagramas de Sequência 2.4 – Diagramas de Atividades 2.5 – Diagramas de Máquina de Estados	6 h-a
UNIDADE 3 – Linguagem de Programação Orientada a Objetos 3.1 – Classes, métodos e atributos 3.2 – Visibilidade de métodos e atributos (pública, privada e protegida) 3.3 – Métodos assessores 3.4 – Sobrecarga de método 3.5 – Sobrecarga de operador 3.6 – Herança 3.7 – Interfaces 3.8 – Tratamento de exceções	10 h-a
UNIDADE 4 – Projeto de Software 4.1 – Análise de Requisitos 4.2 – Projeto Geral 4.3 – Projeto Detalhado 4.4 – Codificação 4.5 – Testes e Validação	16 h-a
TOTAL	36 h-a

AValiação (CONFORME Normas Acadêmicas – RES CD 083/05)

Trabalho Prático Orientado – Projeto de Software

- Parte 1 – Análise de Requisitos e Diagrama de Casos de Uso: 10 pontos
- Parte 2 – Modularização e Projeto Preliminar: 10 pontos
- Parte 3 – Projeto Detalhado com Visões Arquiteturais e Comportamentais: 35 pontos
- Parte 4 – Codificação: 35 pontos
- Parte 5 – Testes e Validação (Apresentação): 10 pontos

ATIVIDADES COMPLEMENTARES

*atividades não computadas na carga horária total que contribuam à melhoria do processo ensino-aprendizagem

BIBLIOGRAFIA BÁSICA

DEITEL, Harvey M.; DEITEL, Paul J. **Java: como programar**. Tradução de Edson Furmankiewicz. 6. ed. São Paulo: Pearson Education do Brasil, 2005. xl, 1110 p., il. ISBN 978-85-7605-019-3 (broch.).

SANTOS, Rafael. **Introdução à programação orientada a objetos usando JAVA**. Rio de Janeiro: Campus, c2003. 319 p., il. (Série Campus - SBC Sociedade brasileira de computação). Inclui sumário e bibliografia. 12ª reimpressão. ISBN 978-85-352-1206-8 (broch.).

SINTES, Anthony. **Aprenda programação orientada a objetos em 21 dias**. São Paulo: Pearson Education do Brasil, 2010. 693 p., il. Inclui índice. ISBN 978-85-346-1461-0 (broch.).

BIBLIOGRAFIA COMPLEMENTAR

FOWLER, Martin, 1963-. **UML Essencial: um breve guia para a linguagem-padrão de modelagem de objetos**. 2. ed. Porto Alegre: Bookman, 2000. 169 p., il. Inclui bibliografia e índice. ISBN 85-7307-729-8 (broch.).

FURLAN, José Davi. **Modelagem de objetos através da UML: the unified modeling language**. São Paulo: Makron Books do Brasil, c1998. xiv, 329 p., il. Inclui bibliografia e índice. ISBN 85-346-0924-1 (broch.).

HUBBARD, John R. **Teoria e problemas de programação em C++**. 2. ed. Porto Alegre: Bookman, 2003. 392 p., il. (Coleção Schaum). Inclui bibliografia e índice. ISBN 85-363-0251-8 (broch.).

LIPPMAN, Stanley B.; LAJOIE, Josée; MOO, Barbara E. **C++ primer**. 4. ed. Boston: Addison-Wesley, 2005. xxi.; 885. ISBN 978-0-201-72148-5.

MONTENEGRO, Fernando; PACHECO, Roberto. **Orientação a objetos em C++**. Rio de Janeiro: Ciência Moderna, c1994. xix, 394 p., il. Inclui bibliografia.

SILVA FILHO, Antonio Mendes da. **Introdução à programação orientada a objetos com C++**. Rio de Janeiro: Elsevier, 2010. xxii.; 283. ISBN 978-85-352-3702-3 (broch.).

PROFESSOR (A) RESPONSÁVEL:	Data
<hr/> Prof. Leandro Resende Mattioli	15/02/2016

COORDENADOR (A) DE CURSO	Data
<hr/>	