

	<div>Universidade Federal do Espírito Santo</div> <div>Centro Tecnológico</div> <div>Departamento de Informática</div>		
PLANO DE ENSINO			
Disciplina: Programação Orientada a Objetos		Código: INF15933	
Curso: Engenharia Elétrica		Período: 2º	
Carga Horária Semanal: 04	Teoria: 02	Exercício: 00	Laboratório: 02
Carga Horária Semestral: 60	Créditos: 04	Período Letivo: 2022/2	

OBJETIVO DA DISCIPLINA
<p>Projetar e programar usando os conceitos de programação orientada a objetos. Aprender a usar linguagens de programação orientadas a objetos.</p>
EMENTA
<p>Princípios do paradigma orientado a objetos. Classes e objetos. Atributos e métodos. Associações entre classes (composição). Construtores e destrutores. Sobrecarga. Modificadores de acesso/visibilidade. Membros de objeto vs. membros de classe. Herança. Sobrescrita. Polimorfismo. Classes e métodos abstratos. Ampliação (upcast) e estreitamento (downcast). Identificação de tipos em tempo de execução. Exceções. Modularização. Classes e métodos genéricos. Estudo aprofundado de uma linguagem de programação orientada a objetos.</p>
PROGRAMA DA DISCIPLINA
<p>Conteúdo Programático:</p> <ol style="list-style-type: none"> 1. Orientação a Objetos (5 horas) <ol style="list-style-type: none"> 1.1. Princípios fundamentais, conceitos básicos e avançados 1.2. Da programação estruturada à programação OO 2. Introdução à programação OO em uma Linguagem (10 horas) <ol style="list-style-type: none"> 2.1. Tipos primitivos 2.2. Variáveis e constantes 2.3. Operadores 2.4. Controle de fluxo 2.5. Entrada e saída de dados básica 3. Classes e objetos (15 horas) <ol style="list-style-type: none"> 3.1. Definição de classes 3.2. Criação e destruição de objetos 3.3. Atributos e métodos 3.4. Vetores 3.5. Atributos de classe 4. Herança (15 horas) <ol style="list-style-type: none"> 4.1. Composição e herança 4.2. Sobrescrita e sobrecarga de métodos 4.3. Polimorfismo e amarração tardia 4.4. Classes e métodos abstratos 4.5. Interfaces

5. Exceções e controle de erros (5 horas)
6. Modularização (5 horas)
 - 6.1. Pacotes
 - 6.2. Importação
 - 6.3. Especificadores de acesso
 - 6.4. Ferramentas jar e javadoc
7. Tópicos avançados (5 horas)
 - 7.1. Enumerações
 - 7.2. Datas
 - 7.3. Expressões lambda
 - 7.4. Estruturas de Dados

METODOLOGIA DE ENSINO

O conteúdo teórico da disciplina será exposto pelo docente utilizando recursos tecnológicos (computador e projetor) e a lousa. Durante as aulas, os alunos realizarão exercícios para colocar em prática os conteúdos apresentados. Além disso, serão propostos trabalhos computacionais para que os alunos tenham a oportunidade de aplicar os novos conhecimentos na resolução de tarefas reais. A interação com os alunos acontecerá de forma presencial e utilizando meios eletrônicos (página do professor e e-mail).

AVALIAÇÕES

Serão utilizados como instrumentos de avaliação provas, trabalhos e exercícios feitos durante as aulas. As provas valerão 60% da nota do semestre, enquanto trabalhos e exercícios valerão 30% e 10%, respectivamente.

A média parcial da disciplina (MP) será calculada usando a seguinte fórmula:

$$MP = 0.2 * P1 + 0.2 * P2 + 0.2 * P3 + 0.15 * T1 + 0.15 * T2 + 0.1 * E$$

onde:

- T1 e T2 são trabalhos com valor de 10 pontos cada.
- P1, P2 e P3 são provas valendo 10 pontos cada.
- E é a pontuação por exercícios realizados em sala com valor total de 10 pontos.

Serão aprovados os alunos que obtiverem $MP \geq 7.0$ e mais de 75% de presença.

Aqueles que não forem aprovados, poderão realizar prova final (PF) valendo 10 pontos e a nota final da disciplina (NF) será dada por:

$$NF = \frac{MP + PF}{2}$$

Serão aprovados os alunos com $NF \geq 5.0$. Aqueles que não alcançarem tal pontuação serão convidados a realizar a disciplina novamente.

BIBLIOGRAFIA

- DEITEL, Paul J.; DEITEL, Harvey M. Java: como programar. 4. ed. Porto Alegre: Bookman, 2003. xx, 1386 p.
- DEITEL, Harvey M.; DEITEL, Paul J. C++: como programar. 5. ed. São Paulo: Editora Pearson, 2006.
- BOOCH, Grady; RUMBAUGH, James; JACOBSON, Ivar. UML: guia do usuário. 2a. edição. Rio de Janeiro: Editora Elsevier, 2012.

BIBLIOGRAFIA COMPLEMENTAR

- SILVA, Ricardo Pereira da. UML 2 em modelagem orientada a objetos. 1. edição. Florianópolis, SC: Visual Books, 2007. 232 p.

- SANTOS, Rafael. Introdução à programação orientada a objetos usando JAVA. 1. edição. Riode Janeiro: Campus, 2003. 319 p.
- STROUSTRUP, Bjarne. Princípios e práticas de programação com C++. 1. edição. Porto Alegre: Bookman, 2012. xxvii, 1216 p.
- HORSTMANN, Cay S. Conceitos de computação com Java. 5. ed. Porto Alegre, RS: Bookman, 2009. xiv, 720 p.
- SCHILDT, Herbert. Java para iniciantes. 5. ed. Porto Alegre, RS: Bookman, 2013. xviii, 614 p.