



MINISTÉRIO DA EDUCAÇÃO
SECRETARIA DE EDUCAÇÃO PROFISSIONAL E TECNOLÓGICA
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA

PLANO DE DISCIPLINA			
IDENTIFICAÇÃO			
CAMPUS: Campina Grande			
CURSO: Bacharelado em engenharia da computação			
DISCIPLINA: Programação orientada a objetos		CÓDIGO DA DISCIPLINA: 34	
PRÉ-REQUISITO: Algoritmos e Programação e Laboratório de algoritmos e programação			
UNIDADE CURRICULAR: Obrigatória [X] Optativa [] Eletiva []		SEMESTRE/ANO: 3º	
CARGA HORÁRIA			
TEÓRICA: 67h.r	PRÁTICA: 0h.r	EaD¹:0h.r	EXTENSÃO:
CARGA HORÁRIA SEMANAL: 4h/a			
CARGA HORÁRIA TOTAL: 67h.r			
DOCENTE RESPONSÁVEL: Daniel Marques Vasconcelos Guimarães			

EMENTA

O paradigma de programação orientada a objetos. Classes e objetos. Troca de mensagens, composição e coleções de objetos. Herança. Sobreposição. Encapsulamento. Visibilidade. Interface e polimorfismo. Sobrecarga. Tratamento de exceções. Desenvolvimento de programas orientados a objetos através de uma linguagem de programação

OBJETIVOS DA DISCIPLINA/COMPONENTE CURRICULAR (Geral e Específicos)
--

Geral

- Apresentar a metodologia de desenvolvimento orientada a objetos, mostrando as técnicas e ferramentas para criação de programas usando uma linguagem de programação orientada a objetos;
- Entender a principal diferença entre programas desenvolvidos utilizando a tradicional metodologia de programação estruturada e orientada a objeto.
- Familiarizar-se com os principais conceitos que determinam o entendimento do paradigma orientado a objeto;
- Valorizar a importância da utilização de boas práticas de programação na elaboração de código fonte.

Específicos

- Instalar o pacote de programas necessário para dar início à prática de programação;
- Configurar o ambiente de desenvolvimento para programação;
- Importar bibliotecas para uso em projetos de programação;
- Escrever programas utilizando dos recursos disponíveis para tratamento de erros e exceções.

1ª unidade

- Fundamentos da Linguagem C++:
 - Histórico e evolução da linguagem C++;
 - Arquitetura da tecnologia C++;
 - Características da linguagem;
 - Produtos e APIs C++;
 - Escrevendo, compilando e executando aplicações C++;
 - Estado da arte em ambientes de desenvolvimento e execução;
 - Abstração, objetos e visão geral de conceitos de POO;
 - Utilização de suporte ferramental adequado e configuração do ambiente de trabalho;
- Programação Orientada a Objetos com C++
 - Classes e criação de objetos;
 - Membros de classe: atributos e métodos (classe e instância);
 - Abstração de dados e encapsulamento;
 - Construtores e suas características;
 - Definindo mensagens e interface de objetos;
 - Sobrecarga e sobreposição de métodos;
 - Ciclo de vida dos objetos (instanciação à destruição);
 - Classes Wrappers (Boolean, Character, Short, Integer, etc.);
 - Estruturação e Manipulação de Objetos em C++;
 - Herança e noções de Polimorfismo.
- Tipos, Literais, Operadores e Controle de Fluxo:
 - Palavras reservadas da linguagem;
 - Constantes e variáveis;
 - Tipos primitivos e de referência;
 - Expressões;
 - Coerção, conversão e promoção de tipos;
 - Operadores: atribuição, aritméticos, relacionais, lógicos e bits;
 - Estruturas de controle de fluxo;
 - Operador '==' versus método equals (Objeto);
 - Enumerações versus Variáveis de Classe.

2ª unidade

- Reutilização com Herança e Composição de Objetos:
 - Quando usar Herança ou Composição;
 - Técnicas de composição e associação de objetos;
 - Herança: vantagens e desvantagens sobre composição;
 - Polimorfismo com herança e com composição;
 - Upcasting e Downcasting;
 - Boas práticas de programação;
 - Padrões de Projeto (essenciais) e boas práticas de programação.
- Interfaces e Polimorfismo:
 - Fundamentos sobre polimorfismo;
 - Aplicando polimorfismo com Interfaces;
 - Classes abstratas e métodos abstratos;
 - Mecanismo Late binding (vinculação dinâmica);
 - Interfaces e Herança múltipla em C++.

3ª unidade

- Encapsulamento e Visibilidade:
 - Definindo e refinando encapsulamento;
 - Modificadores de visibilidade: public, protected, default e private;
 - Criação de pacotes em C++;
 - Importação de classes.
- Tratamento de Erros e Exceções:
 - Fundamentos acerca de tratamentos de erros e seus tipos;
 - Mecanismos Try-Catch e Finally. o Capturando e lançando exceções, finalizando exceções;
 - Exceções padrão em C++;
 - Criando novas exceções;
 - Exceções Runnable.

4ª unidade

- Entrada e Saída Padrão de Dados em C++:
 - Entrada padrão de dados.
 - Saída padrão de dados;
 - Entrada/Saída de dados GUI.
- Arrays e Strings:
 - Arrays simples e multidimensionais;
 - Ordenação de arrays (classe Arrays);
 - Características e manipulação de Strings e caracteres;
- Arquivos e Fluxos de Dados em C++:
 - Manipulação de dados em arquivos;
 - Arquivos, fluxos de entrada e saída em C++;
 - Leitura e gravação de Objetos e Textos em C++.

METODOLOGIA DE ENSINO

Aulas teóricas expositivas, aulas práticas, pesquisas individuais e em grupo, seminários, discussões e listas de exercícios.

RECURSOS DIDÁTICOS

[X] Quadro
 [X] Projetor
 [X] Vídeos/DVDs
 [X] Periódicos/Livros/Revistas/Links
 [] Equipamento de Som
 [X] Laboratório
 [X] Softwares²
 [X] Outros³

CRITÉRIOS DE AVALIAÇÃO

(Especificar quantas avaliações e formas de avaliação – avaliação escrita objetivo, subjetiva, trabalho, seminário, artigo, etc. - para integralização da disciplina/componente curricular, incluindo a atividade de recuperação final.)

Provas escritas, trabalhos práticos, teóricos e listas de exercícios.

ATIVIDADE DE EXTENSÃO⁴

BIBLIOGRAFIA⁵

Bibliografia Básica:

- DEITEL, H.; DEITEL, P. **Java: como programar**. 8. ed. [S.l.]: PearsonBrasil, 2010;
- SANTOS, R. **Introdução à programação orientada a objetos usando Java**. Rio de Janeiro : Elsevier, 2003. ISBN:9788535212068;
- SIERRA, K. **Use a cabeça!**: Java. 2. ed. [S.l.]: Alta Books, 2009.

Bibliografia Complementar:

- LARMAN, C.. **Utilizando UML e padrões: uma introdução à análise e ao projeto orientados a objetos e ao desenvolvimento iterativo**. 3. ed. Porto Alegre: Bookman, 2007. ISBN: 9788560031528;
- CAVALCANTI, V. M. B., RODRIGUES, N. N. **Estruturas de dados lineares básica : Abordagem prática, com implementações em C e Java**. João Pessoa : IFPB, 2015. ISBN: 9788563406613;
- Goodrich, M. T. **Estruturas de dados e algoritmos em Java**. 5. ed. Porto Alegre: Bookman, 2013. ISBN: 9788582600184;
- BASHAM, B., et al. **Use a cabeça : Servlets & JSP**. 1Rio de Janeiro: Alta Books, 2005. ISBN: 9788576082941;
- HALL, M., P.; BROWN, L. **Core Servlets e JavaServer Pages: tecnologias Core**. Rio de Janeiro: Moderna, 2005, ISBN: 8573934328.

OBSERVAÇÕES

(Acréscitar informais complementares ou explicativas caso o docente(s) considere importantes para a disciplina/componente curricular)

- 1 Para a oferta de disciplinas na modalidade à distância, integral ou parcial, desde que não ultrapassem os limites definidos em legislação.
- 2 Nesse item o professor deve especificar quais softwares serão trabalhados em sala de aula.
- 3 Nesse item o professor pode especificar outras formas de recursos utilizadas que não estejam citada.
- 4 Nesse item deve ser detalhado o PROJETO e/ou PROGRAMA DE EXTENSÃO que será executado na disciplina. Observando as orientações do Art. 10, Incisos I, II, III, IV, V, VI, VII e VIII, da Instrução Normativa que trata da construção do Plano de Disciplina.
- 5 Observar os mínimos de 3 (três) títulos para a bibliografia básica e 5 (cinco) para a bibliografia complementar.

Documento assinado eletronicamente por:

■ Daniel Marques Vasconcelos Guimaraes, PROF ENS BAS TEC TECNOLOGICO-SUBSTITUTO, em 01/04/2022 15:03:25.

Este documento foi emitido pelo SUAP em 30/03/2022. Para comprovar sua autenticidade, faça a leitura do QRCode ao lado ou acesse <https://suap.ifpb.edu.br/autenticar-documento/> e forneça os dados abaixo:

Código Verificador: 279025

Código de Autenticação: 17a3cbce90



R. Tranqüilino Coelho Lemos, 671, Dinamérica, CAMPINA GRANDE / PB, CEP 58432-300
<http://ifpb.edu.br> - (83) 2102-6200