

PLANO DE ENSINO

CURSO: Superior de Tecnologia em Análise e Desenvolvimento de Sistemas

SÉRIE: 3º semestre

DISCIPLINA: Programação Orientada a Objetos I

CARGA HORÁRIA SEMANAL: 03 horas/aula

CARGA HORÁRIA SEMESTRAL: 60 horas/aula

I – EMENTA

Histórico. Abstração e encapsulamento. Classes e objetos. Atributos e métodos. Herança e Polimorfismo. Comunicação e associação. Herança simples e múltipla. Hierarquias de classes. Persistência. Interfaces. Inicialização e destruição de objetos. Padrões de interação entre objetos. Aplicações dos conceitos utilizando uma linguagem de programação orientada a objetos (C#).

II – OBJETIVOS GERAIS

Introduzir conceitos básicos sobre orientação a objeto. Capacitar os alunos a implementarem programas orientados a objetos.

III – OBJETIVOS ESPECÍFICOS

Apresentar os princípios da programação orientada a objeto. Consolidar os conceitos de programação baseada em objetos. Compreender os conceitos fundamentais de programação orientada a objetos. Apresentar a linguagem orientada a objeto C#. Capacitar os alunos a implementarem aplicações utilizando o paradigma de programação orientada a objetos em C#.

IV – COMPETÊNCIAS

Desenvolver programas de computador usando os conceitos programação orientada a objetos. Compreender os conceitos fundamentais da orientação a objetos. Abstrair da realidade as funcionalidades a serem implementadas em objetos.

V – CONTEÚDO PROGRAMÁTICO

Histórico da programação orientada a objetos Vantagens e objetivos da OO

Conceitos de POO Abstração e encapsulamento Classes e objetos

Atributos e métodos Herança e Polimorfismo Comunicação e associação

Herança simples e múltipla Hierarquias de classes Persistência

Interfaces



Inicialização e destruição de objetos Padrões de interação entre objetos Visão geral da plataforma Microsoft .NET
Compreensão dos fundamentos da programação em linguagem C# Criação de objetos em C#
Implementação de técnicas de programação orientada a objetos em C#

VI – ESTRATÉGIA DE TRABALHO

As aulas são predominantemente expositivas apoiadas nas diretrizes do plano de ensino. O desenvolvimento do conteúdo se dá com o apoio do material de leitura, exercícios, textos complementares, discussões, chat e ou fórum.

Para estimular e orientar os alunos, o docente da disciplina conta com a participação do tutor presencial e a distância. Com o objetivo de aprofundar o conteúdo programático e o incentivo à pesquisa, o docente pode utilizar recursos como: simulação, jogos, estudos de caso, artigos científicos, trabalhos individuais ou em grupo, palestras, que permitam aos alunos compreenderem na prática a teoria apresentada.

VII – AVALIAÇÃO

A avaliação é um processo desenvolvido durante o período letivo e leva em conta todo o percurso acadêmico do aluno, como segue:

- Acompanhamento de frequência;
- Acompanhamento de nota;
- Desenvolvimento de exercícios e atividades;
- Trabalhos individuais ou em grupo;
- Projeto Integrado Multidisciplinar;
- Estudos disciplinares;
- Atividades complementares.

A avaliação presencial completa esse processo. Ela é feita no polo de apoio presencial no qual o aluno está matriculado, seguindo o calendário acadêmico. Estimula-se a autoavaliação, por meio da autocorreção dos exercícios, questionários e atividades, de modo que o aluno possa acompanhar sua evolução e rendimento escolar, possibilitando, ainda, a oportunidade de melhoria contínua por meio de revisão e feedback.

VIII – BIBLIOGRAFIA

BÁSICA

MANZANO, J. A. N. **Estudo dirigido de Microsoft Visual C# Express 2013**. Erica, 2014.

SHARP., and John. **Microsoft Visual C# 2013 - Série Passo a Passo**. Bookman, 2015.

LIMA, A. S. **UML 2.5 - Do Requisito à Solução**. 5.ed. São Paulo:Érica, 2014.



COMPLEMENTAR

SCHACH, S. **Engenharia de software**: os paradigmas clássico e orientado a objetos. 7.ed. ArtMed, 2010.

DE LA TORRE. C.; CARMONA, D. **.NET Technology guide for business applications**. Microsoft Press, Washington, 2013. Disponível em <http://aka.ms/NETTechGuide/ebook> acesso em 03/02/2014.

AGUILAR, L. J. **Fundamentos de Programação**: algoritmos, estruturas de dados e objetos. 3.ed. AMGH, 2011.

MARTIN, R. **Princípios, padrões e práticas ágeis em C#**. Bookman, 2011.

LARMAN., and Craig. **Utilizando UML e Padrões - Uma Introdução À Análise e ao Projeto Orientados a Objetos e Desenvolvimento Iterativo**. Bookman, 2007.

