

**ULBRA****UNIVERSIDADE LUTERANA DO BRASIL****Pró-Reitoria de Graduação
Direção Geral de Ensino****CURSO: SISTEMAS DE INFORMAÇÃO****ANO/SEMESTRE
2014/2****DISCIPLINA: Paradigmas de Linguagem de Programação
CODIGO: 204613
PROFESSOR: Márcio Daniel Puntel****CRÉDITOS: 04
C/H TOTAL: 68****PLANO DE ENSINO-APRENDIZAGEM****1. EMENTA**

A disciplina de Paradigmas de Linguagens de Programação proporciona o estudo das características das linguagens de programação (conceitos, tipos de dados, escopo de declarações) e conceitualização dos paradigmas de linguagens de programação (características, facilidades e problemas).

2. OBJETIVOS DA DISCIPLINA**2.1 GERAL:**

O objetivo da disciplina é capacitar o aluno a compreender os diferentes paradigmas de linguagens existentes, assim como, as principais características e peculiaridades das linguagens de programação.

ESPECÍFICO (S)

- Desenvolver no aluno o senso crítico na escolha da melhor linguagem de programação.
- Considerar os aspectos relevantes do problema a ser resolvido e a resolução ou estratégia de resolução adotada.
- Generalizar e abstrair as principais características de cada paradigma de linguagem.
- Desenvolver no aluno a competência de abstração conceitual para o estudo e abordagem de novas linguagens de programação.

3. CONTEÚDO PROGRAMÁTICO

- INTRODUÇÃO ÀS LINGUAGENS DE PROGRAMAÇÃO:
 - a) Conceituação;
 - b) Histórico;
 - c) As linguagens de programação no processo de desenvolvimento de software;
 - d) Abstração: dados, controle, unidade;
 - e) Amarração;
 - f) Unidades de programa.
- ESTRUTURAÇÃO DE LINGUAGENS:
- TIPOS DE DADOS:
 - a) Tipos definidos pelos usuários;
 - b) Tipo abstrato de dados.
- ESTRUTURAS DE CONTROLE E COMPOSIÇÃO DE PROGRAMAS:
 - a) Tipos de estruturas de controle;
 - b) Composição de programas;

- **PARADIGMA ORIENTADO A OBJETOS:** Conceitos do paradigma de orientação a objetos; Princípios da programação orientada a objetos e estruturas; Características das linguagens orientadas a objetos.
- **PARADIGMA FUNCIONAL:** Conceitos do paradigma funcional; Princípios da programação funcional; Características das linguagens funcionais.
- **Visão Geral dos demais principais paradigmas de programação:**
 - a) Paradigma Programação em Lógica;
 - b) Paradigma Concorrente;
 - c) Paradigma Orientado a Eventos;
 - d) Paradigma Distribuído;
 - e) Paradigma Orientado a Agentes;
 - f) Paradigma Orientado a Aspectos.

4. METODOLOGIA

As atividades de aprendizagem consistem em leituras, reflexões e discussões sobre elas, exercícios teóricos e práticos, trabalhos de grupo, e avaliação G1 e G2.

a- Leituras: as leituras indicadas na disciplina deverão ser cumpridas no prazo estabelecido a fim de facilitar a aprendizagem do grupo e as discussões sobre os temas que requerem a participação de cada um dos alunos.

b- Reflexões e discussões: Após as leituras espera-se que os alunos identifiquem os conceitos-chave, ideias e questões. Esses itens formarão a base para a discussão. Os alunos poderão ampliar as questões que estão sendo discutidas e sugerir leituras complementares.

c- Exercícios teóricos e práticos: Exercícios planejados para complementar, ampliar e organizar a aprendizagem. Serão disponibilizados ao longo do semestre acompanhados de data de entrega. Estes exercícios poderão ser individuais ou em grupo conforme a orientação determinada.

d- Estudos de caso: Narrativas de situações que deverão ser exploradas criticamente. Objetivam a interpretação e aplicação prática das discussões teóricas da disciplina.

e- Trabalhos em grupo: Formação de grupos para participação de discussões, execução das tarefas e envolvimento em atividades de grupo e simulações. A formação da equipe e as diretrizes para a formação dos grupos será dada no momento apropriado.

5. PROCESSOS AVALIATIVOS

A avaliação será realizada mediante prova individual e trabalhos desenvolvidos em aula e atividades extra-classe.

G1: Prova individual (70%) e Trabalhos (30%).

G2: Prova individual (50%), Trabalhos (30%) e Seminário (20%).

Trabalhos: Realizar atividades relacionadas às áreas de conhecimento para o projeto.

Seminário: Apresentação dos Trabalhos – realizados no semestre - dos grupos.

- APROVAÇÃO: para aprovação na disciplina o aluno deve ter nota final mínima de 6,0 (seis) ao final do semestre.

- PLÁGIO: em caso de identificação de plágio nos trabalhos, o respectivo trabalho receberá nota 0 (zero).

- FREQUÊNCIA: a presença do aluno em aula é obrigatória, faltas acima de 25% (vinte e cinco por cento) das aulas implicam em falta de frequência (reprovação), independentemente dos demais conceitos.

6. BIBLIOGRAFIA BÁSICA

1. SEBESTA, R. W. **Conceitos de Linguagens de Programação**. Editora Bookman, 5. ed, 2003.
2. MELO, A. C. V. de. **Princípios de linguagens de programação**. São Paulo: Edgard Blücher, 2003.
3. VAREJÃO, F. M. . **Linguagens de Programação - Conceitos e Técnicas**. 1. ed. Rio de Janeiro: Elsevier Editora Ltda (Editora Campus), 2004.

7. BIBLIOGRAFIA COMPLEMENTAR

1. SANTOS, Rafael. **Introdução à programação orientada a objetos usando Java**. Rio de Janeiro: Campus, 2003.
2. MONTENEGRO, Fernando. **Orientação a objetos em C++**. Rio de Janeiro: Ciência Moderna, 1994.
3. TUCKER, Allen e NOONAM, Robert. **Linguagens de Programação - Princípios e Paradigmas**. Editora McGrawHill.
4. FRIEDMAN Daniel. P., HAYNES Christopher T. e WAND Mitchell. **Fundamentos de linguagem de programação** - Segunda Edição. Editora Berkeley, 2001.
5. WATT, David. **Programming Language Concepts and Paradigms**. Prentice Hall International Series in Computer Science, 1990.

10. BIBLIOTECA PERSON (livros disponíveis na biblioteca virtual mediante login no autoatendimento da ULBRA):

1. ASCENCIO, A. F. G, CAMPOS, E. A. V. **Fundamentos da Programação de Computadores: algoritmos, Pascal e C/C++ e Java**. São Paulo: Pearson, 2012.

4. CRONOGRAMA

AULA	DESENVOLVIMENTO
1ª AULA 29/07	Apresentação professor; Apresentação do Plano de Ensino; Introdução as Linguagens (histórico, evolução, tipos, sintaxe e semântica).
2ª AULA 05/08	Estudo da estrutura dos tipos de dados (Simples e compostos. Tipos primitivos, estruturados e definidos pelo usuário. Operações entre tipos de dados). Estruturas de Controle (procedimentos, funções, parâmetros).
3ª AULA 12/08	Paradigma Funcional (introdução a lógica funcional)
4ª AULA 16/08	Atividade não presencial – Pesquisar qual linguagem você utilizaria para desenvolver um sistema. Por quê?
5ª AULA 19/08	Linguagem Funcional (conceitos, ambiente, estrutura, características, notação, sintaxe e semântica).
6ª AULA 26/08	Estudo do Paradigma Concorrente.
7ª AULA 02/09	Paradigma Programação em Lógica (histórico, conceitos, características e linguagens). Exercício de Programação em Lógica.
8ª AULA 09/09	(Trabalho G1-01) Estudo de Caso usando linguagem estruturada.
9ª AULA 16/09	Avaliação G1 - Prova
10ª AULA 23/09	Paradigma Orientado a Objetos (histórico, conceitos, características e linguagens). Comparação ente Programação Estruturada e POO

11ª AULA 30/09	Estudo dos conceitos da programação orientada a objetos: classes, objetos, atributos, operações, herança, abstração, representação, encapsulamento, polimorfismo e comunicação entre objetos. Exercícios sobre classe e objetos
12ª AULA 07/10	Paradigma Orientado a Objetos (herança e abstração) Exercícios sobre herança e abstração.
13ª AULA 14/10	Exercícios sobre herança e abstração.
14ª AULA 21/10	(Trabalho G2-01) Estudo de Caso usando POO.
15ª AULA 28/10	Paradigma Orientado a Objetos (polimorfismo e encapsulamento). Exercícios sobre polimorfismo e encapsulamento.
16ª AULA 04/11	Exercícios sobre polimorfismo e encapsulamento.
17ª AULA 11/11	Linguagens para alto desempenho, programação paralela, programação distribuída. Cases.
18ª AULA 18/11	Seminário sobre Paradigma Orientado a Eventos Seminário sobre Paradigma Orientado a Agentes
19ª AULA 25/11	Seminário sobre Paradigma Orientado a Aspectos Seminário sobre Paradigma Distribuído
20ª AULA 02/12	Avaliação de Grau (G2) – Prova
21ª AULA 09/12	Revisão
22ª AULA 16/12	Substituição de Grau – Prova
Feriadados:	