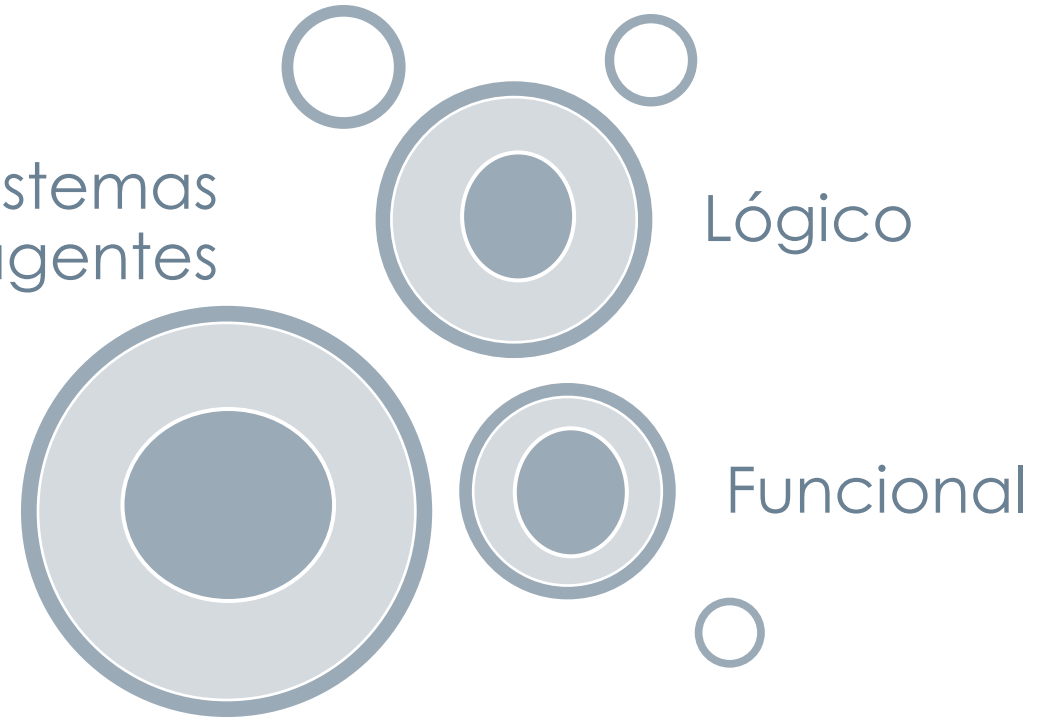


Sistemas
Multiagentes



Paradigmas – Aula 01a

Professora: Milene Serrano

Agenda

- › Considerações Iniciais
- › Plano de Ensino
- › Divisão em Grupos
- › O que esperar do curso?
- › Referências Bibliográficas
- › Cronograma
- › Considerações Finais



Considerações Iniciais

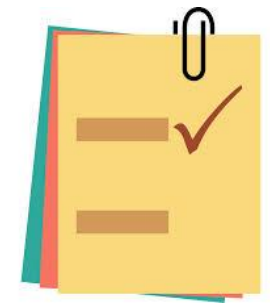
Considerações Iniciais

Conhecer novos paradigmas é algo muito interessante, pois aprendemos novas formas de estruturar nosso raciocínio lógico, novas linguagens de programação e muito mais! :)

Vamos à estrutura do curso!

Plano de Ensino

Ementa



Visão geral sobre Linguagens de Programação;

Definição e caracterização dos principais Paradigmas de Programação, e

Noções preliminares conceituais e de implementação nesses principais Paradigmas de Programação, visando despertar nos alunos de graduação um futuro aprofundamento nos paradigmas e linguagens que mais apreciar/simpatizar.

Objetivos



A disciplina de Paradigmas de Programação visa, principalmente:

- apresentar - i.e. definir e caracterizar - diferentes paradigmas de programação (ex. Imperativo, Funcional, Lógico, Orientado a Objetos, Orientado a Convenções, e outros);
- estabelecer qual o paradigma bem como a linguagem de programação são mais adequados na solução de um problema - análise Problema/Solução; e
- exercitar na prática os paradigmas de programação, implementando soluções para problemas em diferentes domínios cognitivos, usando como base as linguagens mais adequadas para tanto.

Conteúdo Programático



Visão geral sobre Linguagens de Programação (TEORIA):

- Introdução ao Conceito de Paradigma;
- Linguagens de Programação;
- Conceitos Matemáticos em Linguagens de Programação;
- Representação em Linguagens de Programação;
- Hierarquia de Linguagens de Programação;
- Especificação de Linguagens de Programação, e
- Analisadores e Outros em Linguagens de Programação.

Conteúdo Programático

Paradigma Funcional

- Definição e Caracterização;
- Linguagem Haskell, e
- Implementação.



Paradigma Lógico

- Definição e Caracterização;
- Linguagem PROLOG, e
- Implementação.

PROLOG
| ?-

Conteúdo Programático

Paradigma Concorrente

- Definição e Caracterização;
- Linguagem JAVA (Multilinhas) ou ADA ou Modula, e
- Implementação.



Paradigma Multiagentes

- Definição e Caracterização,
- Modelagem, e
- Implementação.



Java Agent DEvelopment Framework

Método de Ensino



Aulas práticas com introdução teórica sobre os principais Paradigmas de Programação e suas respectivas Linguagens de Programação (as mais comumente utilizadas na comunidade da computação), e

Moodle como principal mecanismo de comunicação.

- <http://aprender.unb.br/login/index.php>
- CPF *plus* SENHA

Avaliação



A menção final será constituída das seguintes avaliações:

- **Nota 1 (15% da Nota Final)**, constituída por:
 - Apresentação e entrega do trabalho no Paradigma Funcional (Linguagem Haskell e Plataforma Haskell com GHC e GHCi).
- **Nota 2 (15% da Nota Final)**, constituída por:
 - Apresentação e entrega do trabalho no Paradigma Lógico (Linguagem Prolog e Plataforma LPA_WinProlog).
- **Nota 3 (50% da Nota Final)**, constituída por:
 - Prova sobre os Paradigmas Funcional, Lógico e Concorrente.
 - Escrita à mão, individual, com parte conceitual e programação.
 - **Falta apenas devidamente justificada**, podendo, assim, realizar a SUB para cumprimento da avaliação individual.
- **Nota 4 (20% da Nota Final)**, constituída por:
 - Apresentação e entrega do trabalho no Paradigma de Sistemas Multiagentes (Linguagem Java e Plataforma JADE), com noções do Paradigma Concorrente.

Colas, Cópias e Plágio



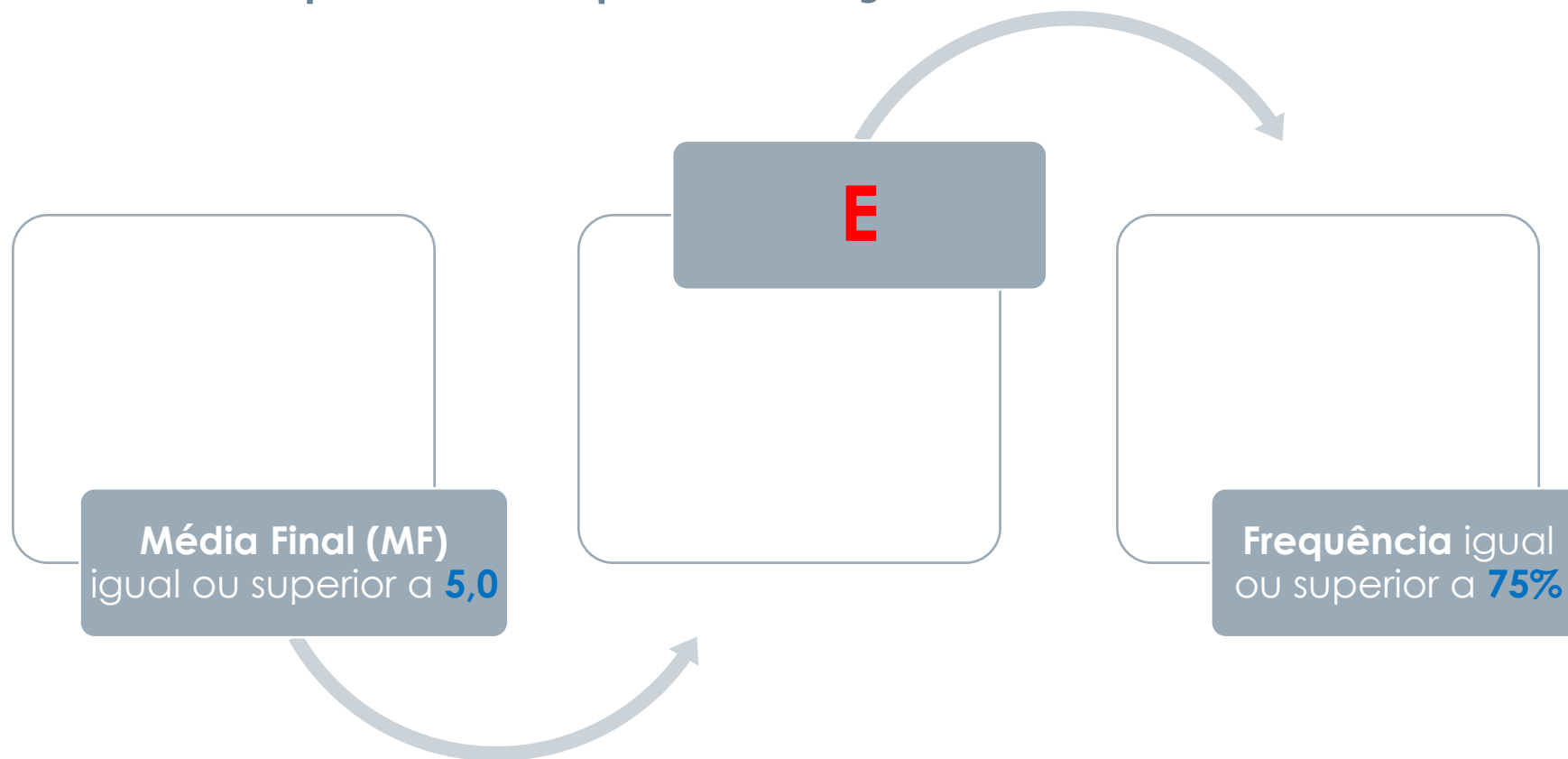
Nas avaliações (sejam atividades, trabalhos ou prova), cópias, colas e/ou plágios receberão nota zero (todos os envolvidos).

Presença nas Aulas

Portanto, compareçam às aulas!
Ok? :)



Critérios para Aprovação



$$MF = (Nota\ 1 * 15\%) + (Nota\ 2 * 15\%) + (Nota\ 3 * 50\%) + (Nota\ 4 * 20\%)$$

Divisão em Grupos

Divisão em Grupos



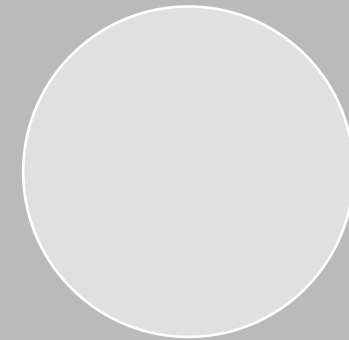
Número de Participantes:

- 3 (ideal), mas dependendo do número de alunos na disciplina, precisamos aumentar esse número.



Objetivos:

- Desenvolvimento de algumas atividades em sala, e
- Trabalhos extras nos diferentes paradigmas, com apresentações, elaboração das respectivas documentações e implementações.



Paradigmas de Programação Envolvidos:

- Todos (Funcional, Lógico, Concorrente e MAS).



O que esperar do curso?

O que esperar do curso?



Principalmente, mas não exclusivamente:

- Conhecimento e capacidade de desenvolver Sistemas em diferentes Paradigmas de Programação, ainda que em um nível preliminar, levando em consideração que para cada problema/desafio existe uma solução mais adequada;
- Aprimoramento do raciocínio lógico e associativo, e
- Possivelmente, o despertar do aluno de graduação para um maior aprofundamento no(s) Paradigma(s) bem como Linguagem(ns) que mais o agradar – em um futuro próximo.

Referências

Referências

Bibliografia Básica

[EBRARY] Scott, M. L. **Programming Language Pragmatics**. eISBN: 9780080515168. 2ª. Edition. 915 pages. Editor: Morgan Kaufmann. Saint Louis, MO, USA. November 2005.

[BIBLIOTECA FGA – 6 Unidades] Tucker, Allen B.; Noonan, Robert. **Linguagens de Programação: Princípios e Paradigmas**. 2ª. Edição. São Paulo: McGraw-Hill, c2009. xxiii, 599 p. ISBN 9788577260447 OU Tucker, Allen B.; Noonan, Robert. **Programming Languages: Principles and Paradigms**. 2ª. Edition. Boston: McGraw-Hill, c2007. xxiii, 600 p. ISBN 9780072866094.

[BIBLIOTECA FGA – 15 Unidades] Cormen, Thomas H. **Algoritmos: Teoria e Prática**. Rio de Janeiro: Elsevier, c2002. 916 p. ISBN 9788535209266.

Referências

Bibliografia Complementar

[OPEN ACCESS] Paradigma Orientado a Convenção sobre Configuração (Híbrido: Estruturado, OO e Funcional) **Grails Platform**: <https://grails.org/documentation.html> e <https://grails.org/wiki/version/Documentation%20Portuguese/9> (Julho 2017).

* Extra, caso queiram conhecer mais um paradigma.

[OPEN ACCESS] Paradigma Funcional. The Haskell Programming Language: <http://book.realworldhaskell.org/read/> e <http://learnyouahaskell.com/chapters> (principais) e <http://www.haskell.org/haskellwiki/Haskell> e <https://www.haskell.org/platform/> e <http://www.haskell.org/haskellwiki/GHC/GHCi> (Julho 2017).

[OPEN ACCESS] Paradigma Lógico. LPA WinProlog: <http://www.lpa.co.uk/index.htm> e http://www.lpa.co.uk/dow_doc.htm (principais LPA WinProlog) OU SWI Prolog: http://www.swi-prolog.org/pldoc/doc_for?object=manual (principal SWI Prolog) e <http://www.swi-prolog.org/> OU GNUProlog/gProlog: <http://www.gprolog.org/#manual> (principal gProlog) e <http://www.gprolog.org/> (Julho 2017).

Referências

Bibliografia Complementar

[OPEN ACCESS] Paradigma Multiagentes (Híbrido: Estruturado, OO e Comportamental) **Jade Documentation**. Multiagent Systems: <http://jade.tilab.com/dl.php?file=JADE-doc-4.3.2.zip> (principal) e <http://jade.tilab.com/>
(Julho 2017).

[OPEN ACCESS] **Introduction to Computer Science Programming Paradigms**. Stanford Graduate School of Education (Stanford University). Stanford, CA. October 2014.
<https://see.stanford.edu/Course/CS107>
http://videlectures.net/stanfordcs107s08_programming_paradigms/
<https://www.udemy.com/cs-107-programming-paradigms/>
(Julho 2017)

Referências

Biblioteca Virtual Ebrary:

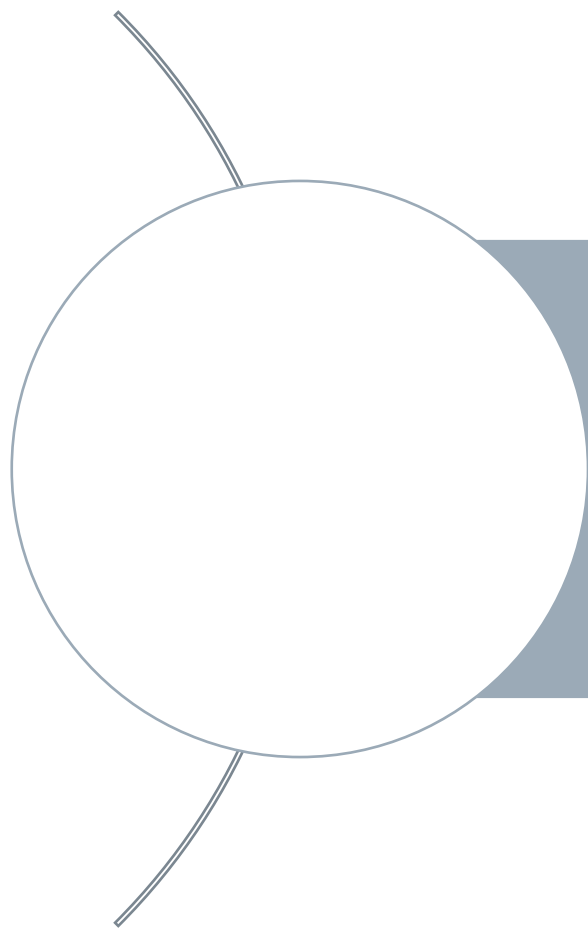
- <http://site.ebrary.com/lib/univbrasil/brasil/home.action>

*Vocês já acessaram?
O acervo de livros virtuais
é muito bom!*



Cronograma

Cronograma



Cronograma

<<disponível no Moodle>>

Considerações Finais

Considerações Finais

- › Nessa aula foi apresentado o plano de ensino.
- › Adicionalmente, foram definidos os grupos de trabalho.
- › Sejam bem vindos!



Dúvidas?

Orientações?

Sugestões?

FIM

mileneserrano@unb.br ou mileneserrano@gmail.com