Apresentação e Plano de Ensino

Prof. Dr. Lucas C. Ribas

Disciplina: Programação Orientada a Objetos

Departamento de Ciências de Computação e Estatística





Agenda



- Apresentação disciplina
 - Objetivo
 - Conteúdo
- Bibliografia
- Avaliação
- Material



Grupo de Pesquisa em IA e Computação Móvel

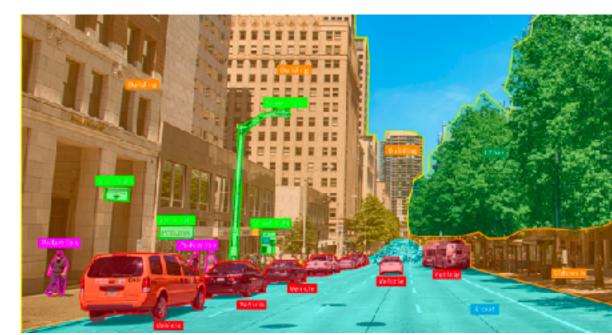


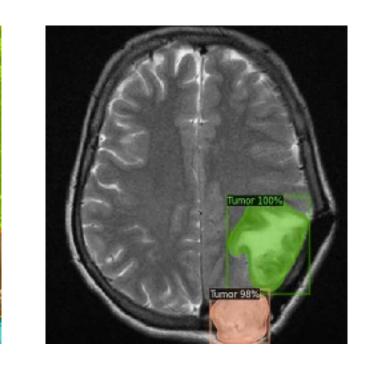
Pesquisa: Inteligência Artificial, Visão Computacional, Reconhecimento de Padrões, Computação Móvel e Aplicações em Biologia, Medicina e Ciência dos Materiais.

Possibilidade de bolsas de IC Institucional e FAPESP

Agende um horário para conhecer o grupo: <u>lucas.ribas@unesp.br</u> (sala 12 - 1º andar - DCCE)









Objetivos



Conhecer os aspectos históricos e os conceitos fundamentais do paradigma de orientação a objetos, dominando os princípios de programação orientada a objetos e como esse paradigma é suportado em linguagens de programação, sendo, finalmente, capaz de implementar sistemas de softwares segundo a visão da filosofia orientada a objetos.



Programa da Disciplina



- Fundamentos do Paradigma de Orientação a Objetos
- Conceitos Básicos de Linguagem Orientada a Objeto
- Classes e Objetos
- Controle de Acesso
- Relacionamento entre objetos
- Herança e Polimorfismo
- Biblioteca de Classes e Coleções
- Conceitos Avançados e Interface Gráfica



Linguagem de Programação



- Java
- Cada tópico terá exemplos usando a linguagem Java
- Será cobrado em prova
- Recomendações:
 - Manter e melhorar as boas práticas de programação:
 - Endentação;
 - Bom uso da memória;
 - Usar convenção de código;
 - Comentários pertinentes;
 - Documentação.



Java



- JDK (Java Development Kit) J2SE. Pode-se encontrar em: http://www.oracle.com/technetwork/java/javase/downloads
- Cuidado, existe também o JRE (Java Runtime Environment), que é feito apenas para rodar programas em Java, e não desenvolve-los.
- Além do JDK que contém o básico, pode-se usar editores como o Netbeans ou Eclipse.
- BlueJ é um programa bastante interessante que pode ajudar no aprendizado da linguagem http://www.bluej.org
 - No site da disciplina há um link para exemplos escritos para BlueJ



Java



- Java JDK 19
- IDE Apache NetBeans





scdefault conf... Q~ Search (%-I) 0 Frojects X Files Services Favorites El Start Page X dataverse dataverse ► Web Pages ▶ ■ RESTful Web Services ▶ Source Packages Show On Startup 2 My NetBeans What's New **NetBeans IDE** ► 📅 Test Packages ► Other Sources Other Test Sources Learn & Discover ► Bependencies Fig. Runtime Dependencies ▶ ■ Test Dependencies Non-classpath Dependencies Take a Tour **Demos & Tutorials** Featured Demo Java Dependencies ► S Project Files Try a Sample Project gradle-build-scan-quickstart Java SE Applications Fig Source Packages [java] Jave and JavaFX GUI Applications Test Location: /Users/geertjanwiclenga/Desktop/gradle-build-scan-quickstart ► S Con Groupla: Java EE & Java Web Applications Artifactid: gradle-build-scan-quickstart C/C++ Applications 3 On pranch "master": in sync with upstream branch. PHP and HTML5 Applications Mobile and Embedded Applications Hello Apache NetBeans All Online Documentation >> () INCUBATOR GlassFish Server × gradle-build-scan-quickstart - /Users/geertjanwielenga/Desktop/gradle-build-scan-quickstart × HV000001: Hibernate Validator 5.0.2.Fina Registered org.glassfish.ha.store.adapter.cache.ShoalBackingStoreProxy for persistence-type = replicated in BackingStoreFa Info: Info: Grizzly Framework 2.4.1 started in: 46ms - bound to [/0.0.0.0:8080] Info: Grizzly Framework 2.4.1 started in: 3ms - bound to [/0.0.0.0:8181] Info: Grizzly Framework 2.4.1 started in: lns - bound to [/0.0.0.0:4848] Info: Grizzly Framework 2.4.1 started in: 11ms - bound to [/0.0.0.0:3700] Info: GlassFish Server Open Source Edition 5.0 (25) startup time : Felix (8,004ms), startup services(1,567ms), total(9,571ms) Info: JTS5014: Recoverable JTS instance, serverId = [100] Loading module services... Background scanning of projects... E (2 more...)





Carga horária



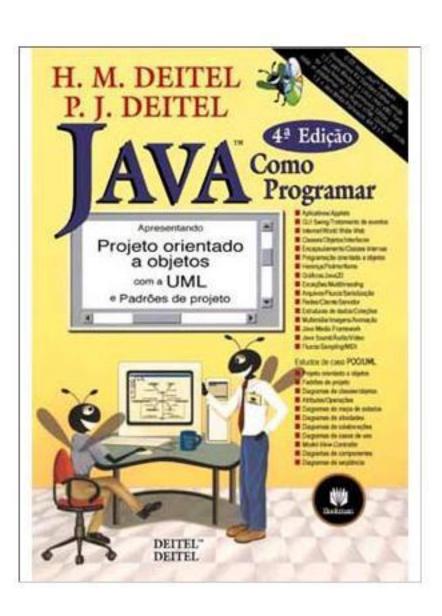
- 60 horas de aulas teóricas: quarta-feira (10-12h) e quinta-feira (14-16h)
- 30 horas para desenvolvimento de trabalhos de implementação (extraclasse). Após as aulas serão disponibilizados exercícios para serem feitos (valem 1 ponto) e entregues. As discussões serão feitas via Classroom.



Bibliografia básica



● DEITEL, H. M. & DEITEL, P.J. "Java: como programar", Bookman, 2003.

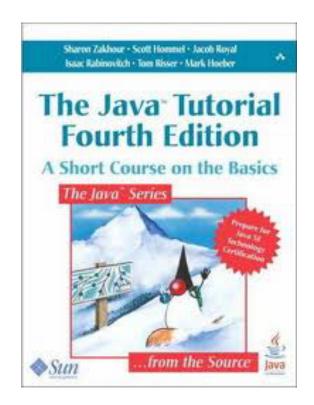




Bibliografia adicional



The Java Tutorials http://download.oracle.com/javase/tutorial/



 Apostila caelum - Java e Orientação a Objetos, disponível em https:// www.caelum.com.br/apostila/apostila-java-orientacao-objetos.pdf



Avaliação



② 2 Provas (Abril e Jun) + Projeto Final + Exercícios

• Média: $M = (P_1 + P_2) \times 5 + (Projeto \times 4) + (Exercícios \times 1) / 10$

- Se M >= 5:
 - Aprovado!
- Senão:
 - Exame final: $nova_M = (M + E) / 2$



Nunca...

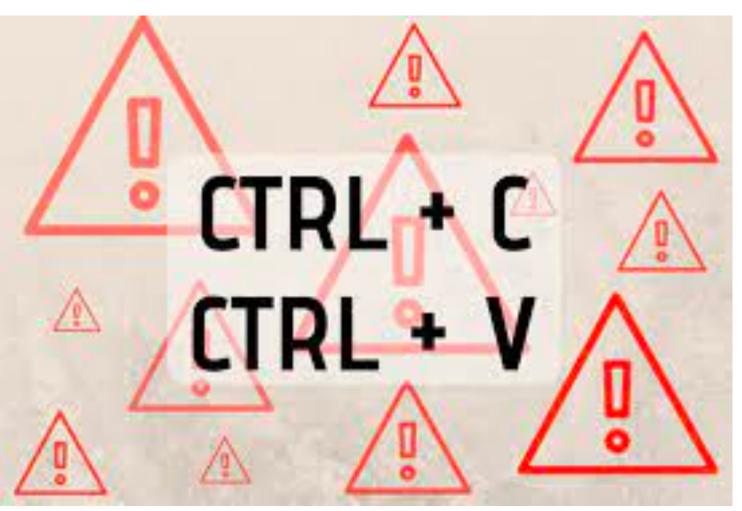


© Copie trabalhos e exercícios…eles serão verificados. Aprenda e faça!











Material e Dúvidas



- Material e Entrega de trabalhos: Google Classroom
 - Código da turma: vdq4g7f
- Contato:
 - E-mail: <u>lucas.ribas@unesp.br</u>
 - Sala 12 1° andar DCCE

