

MC 302A - Segundo semestre de 2011

prof. Fernando Vanini (vanini@ic.unicamp.br)

Programa do Curso

- Motivação: revisão dos conceitos e necessidades em situações reais de programação
- Principais conceitos em programação orientada a objetos
- Estado e comportamento
- Classes e objetos
- Herança, especialização e polimorfismo
- Acoplamento dinâmico
- Hierarquia de classes e sua representação gráfica no padrão UML
- Conceitos adicionais
- Classes abstratas
- Interfaces
- Herança múltipla
- Exceções
- Persistência de Objetos
- Relações entre objetos e sua representação em UML
- associação, agregação e sua representação gráfica no padrão UML
- Coleções e estruturas 'genéricas'
- Recursos para multiprogramação e programação distribuída
- Acesso a meta-informações em tempo de execução
- Aplicação dos conceitos em linguagens atuais (Java, C#, Groovy, Scala, Python, etc.)
- Arquitetura de software e padrões de projetos

Avaliação

- 2 provas teóricas
- ~10 Tarefas de laboratório
- atividades opcionais (a ser definidas no decorrer do curso)

Critério de aprovação

M_p = média aritmética das duas provas

M_a = média aritmética das atividades de laboratório

Se $(M_p < 3.0)$ ou $(M_a < 3.0)$ então $M_f = \min(M_p, M_a)$ senão $M_f = (M_p + M_a) / 2$

Consideração importante: os alunos envolvidos em qualquer tipo de fraude irão para exame com $Media = \min(2.4, M_f)$.

Laboratório

- As aulas de laboratório serão assistidas por um auxiliar de ensino, o Jorge Lima. Na correção dos laboratórios, os critérios considerados serão aos seguintes:
- aderência ao enunciado
- algoritmos usados e sua implementação
- organização e legibilidade do código
- correteza da solução

Exercícios Teóricos

No decorrer do semestre serão propostos exercícios à classe. *A entrega será opcional, e quando oportuno, a solução será comentada em sala.*

Material de Apoio

O material usado em sala (programas exemplo e transparências) e o enunciado dos trabalhos de laboratório serão disponibilizados ao longo do curso estarão disponíveis a partir desta página.

Os programas exemplo mostrados em sala também serão disponibilizados a partir desta página.

Bibliografia

1. The Java Programming Language, Ken Arnold, James Gosling, & David Holmes; Prentice Hall, 4th edition (2005)
2. Thinking in Java, Bruce Eckel; Prentice Hall, 4th edition (2006)
3. Estruturas de Dados e Algoritmos em JAVA, Michael T. Goodrich& Robert Tamassia; Bookman (2007)
4. Data Structures and Algorithms with Object Oriented Design Patterns in C#, Bruno Preiss; <http://www.brpreiss.com/books/opus6/> (2001)
5. Python Tutorial, Guido van Rossum; Python Software Foundation(2008)
6. Groovy in Action, Dirk Koenig et. al.; Manning (2007)
7. Programming in Scala, Martin Oderski et. al.; Artima Press(2007)

8. Padrões de Projeto, Erich Gamma et. al.; Bookman (2002)
9. Guia do Usuário UML, Grady Booch et. al.; Campus(1999)