

Programa de Pós-Graduação em Computação Aplicada

Disciplina: Compiladores e Linguagens de Programação

Semestre: 2013/1

Carga horária total: 45 h-a

Créditos: 03

Área temática: infca

Código da disciplina:

Requisitos de matrícula: inexistente

### **EMENTA**

Conceitos de Linguagens de Programação, enfocando os Paradigmas de Programação e suas especificidades. Categorias de tradutores de linguagens e as fases do processo de tradução. Projeto de Compiladores, discutindo a Análise Léxica e Sintática, a Tradução Dirigida pela Sintaxe, a Verificação de Tipos e a Geração e a Otimização de Código. Ambientes em Tempo de Execução, destacando o Gerenciamento de Memória e o projeto de Máquinas Virtuais. Geradores automáticos de compiladores.

### **CONTEÚDO PROGRAMÁTICO**

1. Introdução às Linguagens de Programação e aos Compiladores
  - 1.1. Evolução das Linguagens de Programação
  - 1.2. Paradigmas de Programação
  - 1.3. Representação de Linguagens de Programação
  - 1.4. Categorias de Tradutores de Linguagens
  - 1.5. Organização de um Compilador e Fases da Compilação
    2. Análise Léxica
      - 2.1. Conceitos Básicos
      - 2.2. Gramáticas Regulares
      - 2.3. Projeto e Geração Automática de Analisadores Léxicos
    3. Análise Sintática
      - 3.1. Conceitos Básicos
      - 3.2. Gramáticas Livres de Contexto
      - 3.3. Análise Sintática top-down
      - 3.4. Análise Sintática bottom-up
      - 3.5. Projeto e Geração Automática de Analisadores Sintáticos
    4. Análise Semântica
      - 4.1. Tradução Dirigida por Sintaxe
      - 4.2. Verificação de Tipos
      - 4.3. Geração de Código Intermediário
    5. Código Final
      - 5.1. Geração de Código Final
      - 5.2. Otimização de Código Final
  6. Ambientes de Tempo de Execução
    - 6.1. Gerenciamento de Memória
    - 6.2. Máquinas Virtuais
  7. Projeto e Implementação de um Compilador
    - 7.1. Especificação da uma Linguagem de Programação

## 7.2. Implementação do Compilador para a Linguagem de Programação

### **BIBLIOGRAFIA BÁSICA**

SEBESTA, R. W. **Concepts of Programming Languages**. Colorado: Addison-Wesley, 2010. 696p.

AHO, A. V. et al. **Compilers: principles, techniques, and tools**. London: Addison-Wesley, 2006. 1000p.

GRUNE, D. et al. **Modern Compiler Design**. Chichester: Wiley, 2000. 736p.

### **AVALIAÇÃO**

A avaliação é baseada em trabalhos que enfocam temas relacionados às Linguagens de Programação e aos Compiladores. Durante a disciplina são realizados no mínimo dois trabalhos. O primeiro aborda um estudo teórico e o aluno produz um artigo enfocando um tema relacionado à disciplina. O segundo trabalho possui um cunho prático (por exemplo, criação de um compilador ou avaliação de implementações existentes) e também deve focar um tópico relacionado a Linguagens de Programação e/ou Compiladores. Outros trabalhos podem ser realizados de acordo com a evolução da disciplina. Todos os trabalhos possuem o mesmo peso na avaliação.

## IDENTIFICAÇÃO

Programa de Pós-Graduação em Computação Aplicada

Disciplina: Análise de Algoritmos

Período: 2013/1

Carga horária total: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006731

Requisitos de matrícula: -

## EMENTA

Apresenta conceitos de modelos de computação e de medidas de complexidade de algoritmos no emprego de programação dinâmica, métodos de busca e ordenação. Aborda também análise de complexidade, algoritmos de tempo polinomial, problemas intratáveis, problemas NP-completo e NP-hard.

## CONTEÚDO PROGRAMÁTICO

Teoria dos conjuntos;

Teoria dos grafos;

Automatos finitos;

Linguagens livre de contexto;

Maquinas de Turing;

Indecidibilidade;

Complexidade;

Problemas NP-Completo e NP-hard.

## BIBLIOGRAFIA BÁSICA

GAREY, M. R.; JOHNSON, D. S. **Computers and intractability**: a guide to the theory of NP-completeness. New York: W. H. Freeman and Company, 1979.

## AVALIAÇÃO

Os alunos desenvolvem o seu aprendizado por meio de trabalhos práticos em laboratórios, leitura e apresentação de artigos selecionados e elaboração de uma monografia.

## **IDENTIFICAÇÃO**

Programa de Pós-Graduação em Computação Aplicada

### **Disciplina: Métodos Matemáticos I**

Período: 2013/1

Carga horária: 45h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006732

Requisitos de matrícula: -

## **EMENTA**

Apresenta temas matemáticos fundamentais nas áreas de simulação e modelagem, fazendo uma revisão dos conceitos básicos de álgebra linear e cálculo diferencial e integral em múltiplas variáveis. Estuda também equações diferenciais ordinárias e parciais, enfatizando sua interpretação e uso em aplicações de modelagem.

## **CONTEÚDO PROGRAMÁTICO**

Álgebra Linear;

Cálculo diferencial e integral em múltiplas variáveis;

Equações diferenciais e parciais aplicadas à modelagem.

## **BIBLIOGRAFIA**

LIMA, E. L. **Álgebra Linear**. Rio de Janeiro: Impa, 1996.

## **AVALIAÇÃO**

Os conceitos da disciplina serão apresentados em aulas expositivas. Assuntos pertinentes ao conteúdo da disciplina são desenvolvidos através de pesquisas e seminários.

Os alunos serão avaliados com trabalhos exigidos em sala de aula e com uma prova final com todo conteúdo abordado na disciplina.

## **IDENTIFICAÇÃO**

Programa de Pós-Graduação em Computação Aplicada

### **Disciplina: Modelagem e Simulação**

Período: 2013/1

Carga horária: 45h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006729

Requisitos de matrícula: -

## **EMENTA**

Apresenta uma introdução à modelagem e simulação de sistemas aplicada ao estudo de casos, abordando métodos de modelagem, processos de geração de modelos, sistemas dinâmicos e processos sequenciais e temporais.

## **CONTEÚDO PROGRAMÁTICO**

Simulação discreta e simulação contínua;

Modelagem de sistemas através de Sistemas de Filas, Redes de Petri e Cadeias de Markov;

Teoria de Sistemas de Filas;

Modelagem através de Equações Diferenciais e algoritmos de simulação;

Linguagens e bibliotecas de rotinas para simulação;

Conceitos básicos de probabilidade e estatística, incluindo regressão;

Aquisição de dados e seleção de distribuição de entrada;

Geração de Números Aleatórios e algoritmos;

Análise estatística dos resultados de simulação;

Comparação entre diferentes opções de projeto e resultados de simulação;

Técnicas de redução de variância;

Projeto experimental;

Metamodelos.

## **BIBLIOGRAFIA**

LAW, A. M.; KELTON, W. D. **Simulation Modeling and Analysis**. 2nd. ed. New York: McGraw-Hill, 1991.

KLEIJNEN, J.; GROENENDAAL, W. **Simulation A Statistical Perspective**. Chichester: John Wiley & Sons, 1992.

### **AVALIAÇÃO**

A avaliação é composta por: (i) apresentação de seminários pelos alunos sobre temas atuais em revistas especializadas na área de simulação. A escolha dos temas é livre. São avaliadas a capacidade de exposição de um tema científico pelo aluno, a forma da apresentação e o domínio do aluno no tema escolhido; (ii) apresentação de ferramentas computacionais na área de simulação, explorando assim possibilidades de ferramentas a serem utilizadas no trabalho final da disciplina; e (iii) prova escrita abordando os conteúdos discutidos em sala de aula.

## **IDENTIFICAÇÃO**

Programa de Pós-Graduação em Computação Aplicada

### **Disciplina: Técnicas de Programação**

Período: 2013/1

Carga horária: 45h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006730

Requisitos de matrícula: -

## **EMENTA**

Apresenta conceitos em técnicas de programação, tratando sobre desenvolvimento de algoritmos, tipos de dados estruturados, conceitos de modularidade e abstração. São apresentados conceitos relacionados aos paradigmas imperativo e orientado a objetos e técnicas de desenvolvimento de software.

## **CONTEÚDO PROGRAMÁTICO**

Paradigma Imperativo;

Arquitetura de von Neumann;

Fluxo de execução;

Controle de fluxo de execução;

Abstrações de dados, procedimentos e funções;

Paradigma Orientado a Objetos;

Modelagem orientada a objetos;

Tipo abstrato de dados, classe;

Encapsulamento, objeto;

Herança, ligação dinâmica e polimorfismo;

Projeto de Software;

Projeto orientado a objetos, UML;

Reuso, desenvolvimento baseado em componentes e padrões de projeto. Framework.

Processo de Software;

Metodologias de desenvolvimento, PSP, TSP, Extreme Programming.

### **BIBLIOGRAFIA**

STROUSTRUP, B. **The C++ programming language**. 2nd. ed. Reading: Addison-Wesley, 1999.

SOMMERVILLE, I. **Software engineering**. 6th. ed. Harlow: Addison-Wesley, 2001.

### **AVALIAÇÃO**

Os conceitos fundamentais de programação são apresentados em aulas expositivas, durante as quais os alunos são incentivados a relacionar o uso destes em situações reais. Assuntos ligados ao projeto e ao processo de software são desenvolvidos através de seminários e debates conduzidos pelos alunos. Durante a disciplina, são propostas atividades extra-classe a serem realizadas com apoio de recursos de laboratório.

## **IDENTIFICAÇÃO**

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Computação Gráfica I**

Período: 2013/1

Carga horária: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006845

Requisitos de matrícula:

## **EMENTA**

Fornece uma visão geral da computação gráfica através de seus fundamentos. Aborda temas como fundamentos de cor e sistemas de cor, síntese e visualização de imagens (câmera sintética), objetos bidimensionais e tridimensionais, modelagem de curvas e superfícies e introdução aos modelos avançados de iluminação.

## **CONTEÚDO PROGRAMÁTICO**

- 1) Introdução. O que é CG? Introdução ao OpenGL e GLUT
- 2) Modelos de Cor. Luz e Sistema Visual Humano
- 3) Pipeline de visualização tridimensional
- 4) Transformações geométricas bi e tridimensionais
- 5) Hardware Gráfico
- 6) Rasterização. Recorte. Processo de Câmera Sintética
- 7) Luz e Iluminação. Realismo
- 8) Modelos de Iluminação Locais e Globais
- 9) Curvas e Superfícies
- 10) Introdução à Animação
- 11) Introdução ao Processamento de Imagens

## **BIBLIOGRAFIA BÁSICA**

GLASSNER, A. **Principles of digital image synthesis**. San Francisco : Morgan-Kaufmann, 1995. v.1 e 2.

FOLEY et al. **Computer graphics: principles and practice**. Reading : Addison-Wesley, 1990. 1174p.

## **AVALIAÇÃO**

A metodologia de ensino empregada consiste:

- exposição dos conceitos, das técnicas e das ferramentas apresentadas na literatura de referência na área;
- realização de estudos de caso;
- realização de trabalhos práticos com o uso de ferramentas e apresentação de seminários

A avaliação dos alunos se dará através de trabalhos individuais, assim como participação e interesse demonstrado nas aulas.

## IDENTIFICAÇÃO

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Inteligência Artificial e Sistemas Inteligentes**

Período: 2013/1

Carga horária: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 006839

Requisitos de matrícula: -

## EMENTA

Apresenta os principais conceitos de inteligência artificial: métodos de resolução de problemas, métodos de representação de conhecimento e inferência automática utilizados na construção de sistemas especialistas. Aborda também tópicos em arquitetura de agentes inteligentes, inteligência artificial distribuída e sistemas multi-agentes.

## CONTEÚDO PROGRAMÁTICO

Introdução à Inteligência Artificial: História, Conceitos, Áreas e Aplicações. Inteligência Artificial Simbólica;

Resolução de problemas: busca em espaço de estados de soluções, heurísticas;

Raciocínio baseado em casos e no uso de analogias (*CBR – Case-Based Reasoning*);

Raciocínio baseado em fatos e regras (*Rule-Based Systems*);

Sistemas Especialistas: conceitos, linguagens, funcionamento e implementação;

Redes Semânticas, *Frames* e *Scripts*;

Ontologias;

Linguagens para o desenvolvimento de programas de Inteligência Artificial.

Sistemas Multi-agentes: agentes inteligentes, reativos, deliberativos; híbridos; arquitetura de agentes (e.g. BDI), linguagens para representação e troca de conhecimentos entre Agentes (e.g. KQML, AgentSpeak-L);

Computação Afetiva: inferência de emoções do usuário, expressão de emoção em máquina e síntese de emoções.

### **BIBLIOGRAFIA**

NILSSON, N. J. **Artificial intelligence**: a new synthesis. San Mateo: Morgan Kaufmann Publishers, 1998.

RUSSEL, R.; NORVIG, P. **Artificial intelligence**: a modern approach. Englewood Cliffs: Prentice Hall, 1995.

### **AVALIAÇÃO**

Exposição de conceitos, técnicas e ferramentas utilizadas na solução de problemas na área; realização de atividades de construção de sistemas inteligentes; exercícios práticos de utilização de ferramentas; apresentação de seminários.

## IDENTIFICAÇÃO

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Pesquisa Operacional**

Período: 2013/1

Carga horária: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 007129

Requisitos de matrícula: -

## EMENTA

Aborda técnicas e aplicações utilizadas pela Pesquisa Operacional para a solução de problemas, apresentando técnicas e aplicações como programação linear, programação inteira, grafos e redes, programação dinâmica, programação não linear e otimização combinatória.

## CONTEÚDO PROGRAMÁTICO

Introdução à Pesquisa Operacional;

Conceitos Básicos;

Programação Linear;

Programação Inteira;

Grafos e redes;

Programação dinâmica;

Introdução à Otimização combinatória.

## BIBLIOGRAFIA

HILLIER, F. S. LIEBERMAN, G. J. **Introduction to operations research**. 6th. ed. New York: McGraw-Hill, 1997.

TAHA, H. A. **Operations research: an introduction**. 6th. ed. Englewood Cliffs: Prentice-Hall, 1997.

## AVALIAÇÃO

Trabalhos de pesquisa sobre aplicações das técnicas de Pesquisa Operacional;

Seminários sobre os diferentes tipos de Programação Matemática;  
Implementação computacional das técnicas de Pesquisa Operacional;  
Avaliações teóricas.

## IDENTIFICAÇÃO

Programa de Pós-Graduação em Computação Aplicada

### Disciplina: Sistemas Distribuídos

Período: 2013/1

Carga horária: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 007135

Requisitos de matrícula: -

## EMENTA

Aborda sistemas de *software* que apresentam distribuição de controle e/ou dados e os algoritmos distribuídos que os fundamentam. Apresenta paradigmas de comunicação e algoritmos distribuídos e discute exemplos clássicos de sistemas distribuídos: sistemas operacionais distribuídos, sistemas de alta disponibilidade, objetos distribuídos, bases de dados distribuídas e simulação distribuída.

## CONTEÚDO PROGRAMÁTICO

- Sistemas distribuídos;
- Algoritmos distribuídos;
- Paradigmas de comunicação em algoritmos;
- Sistemas Operacionais.

## BIBLIOGRAFIA

COLOURIS, G.; DOLLIMORE, J.; KINDBERG, J. **Distributed systems: concepts and design**. 3rd. ed. Reading: Addison Wesley, 2000.

HAROLD, E. R. **Java Network Programming**. 2nd. ed. Cambridge: O'Reilly, 2000.

LYNCH, N. A. **Distributed algorithms**. San Mateo: Morgan Kaufmann, 1996.

TANENBAUM, A. S.; VAN STEEN, M. **Distributed Systems: principles and paradigms**. 1st. ed. New York: Prentice Hall, 2002.

VERÍSSIMO, P.; RODRIGUES, L. **Distributed Systems for Systems Architects**. Norwell: Kluwer, 2001.

## **AVALIAÇÃO**

Apresentação de seminários.

Avaliação escrita.

### **IDENTIFICAÇÃO**

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Automação Industrial**

Período: 2013/1

Carga horária: 45 h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 007121

Requisitos de matrícula: não tem

### **EMENTA**

Introduz os principais conceitos de automação industrial através da filosofia da teoria de grupo e manufatura celular, CAD, CAM e CIM até os sistemas de manufatura flexíveis. Aborda os problemas de seleção de partes, de carregamento e de escalonamento com restrições e heurísticas construtoras e de busca local.

### **CONTEÚDO PROGRAMÁTICO**

1- Tecnologia de Grupo.

- 2.1 Introdução
- 2.2 Histórico
- 2.3 Integração CAD/CAM
- 2.4 Estruturas Básicas de Classificação e Codificação de Partes

2 - Teoria de "Cluster"

- 2.5 Família de Partes
- 2.6 Formulações Básicas
- 2.7 Teoria de Redes
- 2.8 Aplicações em Linhas de Produção

3 – Manufatura Celular

- 3.1 Definição
- 3.2 Modelo de Controle Hierárquico
- 3.3 Controle Computacional da Manufatura – CAD/CAM
- 3.4 CNSs e DNCs

4- Automação

- 4.1 Aplicações
- 4.2 Categorias
- 4.3 Estratégias

#### 4.4 Sistemas de Produção Automatizados

#### 5 - Tecnologias Computer-Aided

##### 5.1 CAD: módulos e hardware

##### 5.2 CAM

###### 5.2.1 Célula de Manufatura

###### 5.2.2 Robôs

###### 5.2.2.1 Definições

###### 5.2.2.2 Configurações

###### 5.2.2.3 Tipos de Robôs

#### 6 - Sistemas de Manufatura Flexíveis

##### 6.1 Introdução

##### 6.2 Definição do Sistema

##### 6.3 Flexibilidade

##### 6.4 Modelo Conceitual

##### 6.5 Problema de Seleção de Partes

##### 6.6 Problema de Escalonamento

#### 7 - Utilização de metaheurísticas

### **BIBLIOGRAFIA BÁSICA**

BEDWORTH, D. D.; HENDERSON, M. R.; WOLFE, P. M. **Computer-Integrated Design and Manufacturing**. New York: McGraw-Hill, 1991.

DORF, R. C.; KUSIAK, A. **Handbook of Design, Manufacturing and Automation**. New York: John Wiley & Sons, 1994.

GROOVER, M. P. **Automation, Production Systems, and Computer-Integrated Manufacturing**. Englewood Cliffs: Prentice Hall, 2001.

JHA, N. K. **Handbook of Flexible Manufacturing Systems**. San Diego, California: Academic, 1991.

KUSIAK, A. **Intelligent Design and Manufacturing**. Chichester: John Wiley & Sons, 1992.

KUSIAK, A. **Concurrent Engineering Automation, Tools and Techniques**. New York: John Wiley & Sons, 1993.

### **AVALIAÇÃO**

- Realização de Seminários pelo corpo docente.

- Realização de provas escritas.
- Realização de estudo dirigido.

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Sistemas Adaptativos Inteligentes**

Período: 2013/1

Carga horária: 45h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 007133

Requisitos de matrícula:

**EMENTA**

Apresenta sistemas adaptativos e técnicas de aprendizado automático, estudando métodos de aprendizado e otimização: algoritmos genéticos, sistemas fuzzy adaptativos e redes neurais artificiais. Estuda também aprendizado simbólico e sistemas híbridos.

**CONTEÚDO PROGRAMÁTICO**

- 1) Aprendizado de máquinas indutivo: conceito de aprendizado supervisionado e não supervisionado;
- 2) Aprendizado supervisionado: generalização de conceitos e o problema do *overfitting*;
- 3) Avaliação do desempenho de algoritmos de aprendizado de máquina: estimativa de erro e precisão de um classificador, validação cruzada (*cross-validation*), validação com *N-Folds*, matriz de confusão;
- 4) Conceitos, algoritmos e simulação de Redes Neurais Artificiais (RNAs). Aplicações práticas de RNAs;
- 5) Conceitos, algoritmos e simulação de Algoritmos Genéticos (AGs). Aplicações práticas de AGs;
- 6) Conceitos, algoritmos e indução de Árvores de Decisão a partir de problemas práticos;
- 7) Conceitos sobre: Sistemas Nebulosos Adaptativos (*Adaptive Fuzzy Inference Systems - Adaptive FIS*), Aprendizado por Reforço (*Reinforcement Learning*), Aprendizado em Redes Bayesianas (*Bayesian Learning*), Aprendizado em Agentes e Sistemas MultiAgente (MAS), SVM (*Support Vector Machines*).
- 8) Técnicas de aprendizado simbólico: Indução de Árvores de Decisão e ILP (*Inductive Logic Programming*);
- 9) Sistemas Inteligentes Híbridos (HIS);
- 10) Utilização prática de ferramentas de aprendizado de máquina (e.g.: SNNS, C4.5, Weka);
- 11) Desenvolvimento de aplicações práticas usando ferramentas de aprendizado de máquinas;
- 12) Tendências e perspectivas na área de sistemas adaptativos (*machine learning*).

### **BIBLIOGRAFIA BÁSICA**

HAYKIN, S. **Neural Networks**: a comprehensive foundation. 2nd. ed. Upper Saddle River, NJ: Prentice Hall, 1999.

MITCHELL, T. M. **Machine learning**. New York: McGraw-Hill, 1997.

MITCHELL, M. **An introduction to genetic algorithms**. Cambridge, Mass: MIT, 1996.

QUINLAN, J. R. **C4.5**: programs for machine learning. San Mateo: Morgan Kaufmann, 1993.

REZENDE, S. O. **Sistemas inteligentes**: fundamentos e aplicações. São Paulo: Manole, 2003.

### **AVALIAÇÃO**

Os alunos serão avaliados considerando-se: (i) sua participação ativa e interesse demonstrado nas aulas expositivas e práticas; (ii) seu desempenho referente à compreensão, análise crítica e apresentação de seminários sobre artigos e trabalhos do estado-da-arte na área de aprendizado de máquina; (iii) os resultados e a análise do desempenho obtido no desenvolvimento de um estudo de caso prático que envolva a aplicação de técnicas de aprendizado de máquina em problemas reais.

Programa de Pós-Graduação em Computação Aplicada

**Disciplina: Métodos Formais para Engenharia de Software**

Período: 2013/1

Carga horária: 45h-a

Créditos: 3

Área temática: Infca

Código da disciplina: 104173

Requisitos de matrícula: -

### **EMENTA**

Principais métodos formais que podem ser aplicados na especificação, refinamento e verificação de sistemas de *software* e no tratamento formal de concorrência de tempo. Critérios de pertinência e aplicabilidade destes métodos aos vários processos de Engenharia de *Software*. Métodos formais baseados em linguagens de especificação funcionais, lógicas, algébricas e baseadas em modelos, incluindo, entre outras: Z, VDM, OBJ, LTL, CTL e PCTL. Métodos de modelagem de programas e processos baseados em sistemas de transição e lógicas temporais. Utilização de verificadores de modelos para provar propriedades de sistemas computacionais.

### **CONTEÚDO PROGRAMÁTICO**

1. Introdução
  - 1.1. Estilos de Especificações
  - 1.2. Métodos Formais nos Processos de Engenharia de Software
  - 1.3. O Papel da Especificação Formal
  - 1.4. Possibilidades de Verificação Formal
2. Noções de Lógica de Predicados, Teoria dos Conjuntos e Matemática Discreta
3. Técnicas de Especificação Formal
  - 2.1. Especificação baseada em modelos: Z e VDM
  - 2.2. Especificações algébricas: OBJ
  - 2.3. Especificações lógicas: LTL, CTL e PCTL
4. Casos de Estudo em Especificação Formal
5. Projeto de Especificação Formal de Software
6. Provadores de Teoremas e Verificação Formal
7. Teste e Verificação de Sistemas com base em Especificação Formal

### **BIBLIOGRAFIA BÁSICA**

ALAGAR, V. S.; PERIYASAMY, K. **Specification of Software Systems**. New York: Springer, 1998.

BAIER, C.; KATOEN, J. P. **Principles of Model Checking**. Cambridge: MIT, 2007.

BERARD, B. et al. **Systems and Software Verification: model-checking techniques and tools**. Berlin: Springer-Verlag, 2001.

BJORNES, D.; JONES, G. B. **The Vienna Development Method: the meta-language**. Berlin: Springer-Verlag, 1978.

MONIN, J. F.; HINCHEY, M. G. **Understanding Formal Methods**. London: Springer-Verlag, 2003.

POTTER, B.; SINCLAIR, J.; TILL, D. **An Introduction to Formal Specification and Z**. 2nd. ed. New Jersey: Prentice Hall, 1996.

WOODCOCK, J.; LOOMES, M. **Software Engineering Mathematics**. London: Addison-Wesley, 1998.

### **AVALIAÇÃO**

O processo de avaliação da disciplina será dividido em duas fases. A primeira fase corresponde ao ensino teórico-prático dos conceitos e técnicas de especificação e verificação formal de sistemas. Nesta fase a avaliação será conduzida através de dois instrumentos: (a) trabalhos teórico-práticos realizados em equipe, com a posterior análise e discussão dos resultados destes trabalhos em aula, e (b) provas individuais. O instrumento (a) terá o peso de 40% e o instrumento (b) terá o peso de 60%.

A segunda fase corresponde à aplicação dos conceitos e técnicas vistos na primeira fase, na elaboração de uma especificação formal para um sistema computacional e na verificação formal deste sistema. Nesta fase a avaliação será conduzida através do acompanhamento e a orientação dos estudantes no desenvolvimento de um projeto de especificação formal de um sistema de computação, com a verificação formal das propriedades do sistema. O projeto será elaborado e desenvolvido em equipe. A segunda fase se encerra pela geração de um artigo técnico-científico e pela apresentação final em aula do artigo. Além da qualidade do projeto final, e das habilidades de comunicação oral e escrita, também serão avaliadas no decorrer do projeto as habilidades de solução de problemas e gerenciamento do projeto.