

Plano de Ensino

1) Identificação

Disciplina: INE5622 - Introdução a Compiladores
Turma(s): 06238
Carga horária: 72 horas-aula Teóricas: 54 Práticas: 18
Período: 1º semestre de 2019

2) Cursos

- Sistemas de Informação (238)

3) Requisitos

- INE5609 - Estruturas de Dados

4) Ementa

Gramáticas, Autômatos e Linguagens; Análise Léxica; Análise Sintática; Análise Semântica e Geração de Código.

5) Objetivos

Geral: Conhecer o processo de especificação e implementação de linguagens de programação, a partir dos conceitos, modelos, técnicas e ferramentas que compõem a Teoria das Linguagens Formais e a Teoria de Compiladores.

Específicos:

- Adquirir uma visão geral sobre o Processo de Compilação sob o ponto de vista de implementação.
- Adquirir noções básicas sobre a Teoria das Linguagens Formais.
- Saber especificar aspectos léxicos e sintáticos de linguagens através de autômatos e gramáticas.
- Conhecer critérios e características usados no projeto e na avaliação de Linguagens de Programação.
- Conhecer as principais técnicas e ferramentas de apoio usadas na construção de compiladores, sabendo usá-las na especificação e implementação de linguagens de programação.
- Obter Subsídios que permitam um melhor entendimento, utilização e avaliação das Linguagens de Programação.

6) Conteúdo Programático

- 6.1) Introdução [6 horas-aula]
 - Compiladores
 - Teoria das Linguagens Formais
- 6.2) Gramáticas [10 horas-aula]
 - Motivação
 - Definição formal
 - Derivação e redução
 - Linguagens
 - Tipos de gramáticas e linguagens
 - Sentença vazia
- 6.3) Linguagens de Programação [4 horas-aula]
 - Conceitos fundamentais
 - Paradigmas
 - Critérios para avaliação e projeto
 - Especificação
 - Ferramentas de implementação
- 6.4) Análise Léxica [10 horas-aula]
 - Autômatos Finitos Determinísticos e Não-Determinísticos
 - Determinização e Minimização
 - Relação entre AF e GR
 - Conjuntos regulares e Expressões Regulares

- Aspectos Léxicos de Linguagens de Programação
- Especificação e implementação de um Analisador Léxico
- 6.5) Análise Sintática [12 horas-aula]
 - Gramáticas Livre de Contexto (GLC)
 - Árvore de derivação, ambigüidade, transformações e simplificações
 - Técnicas de Análise Sintática Ascendentes e Descendentes
 - Aspectos Sintáticos de Linguagens de Programação
 - Especificação e implementação de um Analisador Sintático
- 6.6) Análise Semântica [16 horas-aula]
 - Aspectos semânticos de Linguagens de Programação
 - Modelos de Especificação
 - Regras e Ações Semânticas
 - Especificação e implementação de um Analisador Semântico
- 6.7) Geração/Interpretação de código intermediário [14 horas-aula]

7) Metodologia

Aulas expositivas teóricas e de exercícios e aulas práticas em laboratório para desenvolvimento dos conteúdos teóricos ministrados.

8) Avaliação

Será considerado aprovado na disciplina o aluno que obtiver Nota Final (NF) igual ou superior a 6.0 e frequência igual ou superior a 75%.

A avaliação se dará pela aplicação de 3 provas individuais e 3 trabalhos práticos.

A nota final será composta da seguinte forma:

$$NF = (P1+P2+P3)/3 * 0,6 + (IC1+IC2+IC3)/3 * 0,4.$$

Onde P1, P2 e P3 são as provas, e IC1, IC2 e IC3 referem-se a implementação do compilador (níveis léxico - IC1, sintático - IC2, e semântico - IC3, respectivamente).

Conforme parágrafo 2º do artigo 70 da Resolução 17/CUn/97, o aluno com frequência suficiente (FS) e média final no período (**MF**) entre 3,0 e 5,5 terá direito a uma nova avaliação ao final do semestre (**REC**), sendo a nota final (**NF**) calculada conforme parágrafo 3º do artigo 71 desta resolução, ou seja: $NF = (MF + REC) / 2$.

9) Cronograma

Provas:

* Prova 1 (P1): 24/04/2019;

* Prova 2 (P2): 29/05/2019;

* Prova 3 (P3): 03/07/2019.

Implementação de Compilador:

* Implementação de Compilador 1 (IC1): 25/04/2019;

* Implementação de Compilador 2 (IC2): 30/05/2019;

* Implementação de Compilador 3 (IC3): 10/07/2019.

Prova de Recuperação (REC): 11/07/19

10) Bibliografia Básica

- Notas de aula
- FURTADO, O. J. V. Apostila de Linguagens Formais e Compiladores, versão 2 UFSC, 2002, disponível em www.inf.ufsc.br/~olinto
- AHO, A.V.; SETHI, R. ULLMAN, J.D. Compiladores – Princípios, Técnicas e Ferramentas, Ed. Addison Wesley 2008 / LTC, 1995.

11) Bibliografia Complementar

- FISHER, C. N., CYTRON, R. K., LeBLANNC Jr., R. J.. Crafting a Compiler. Ed. Addison Wesley, 2009.
- SEBESTA, R.W., Conceitos de Linguagens de Programação, ed. Bookman, 5. edição, 2003.
- RAMOS, M. V. M., NETO, J. J., VEGA, I. S., Linguagens Formais: Teoria, Modelagem e Implementação. Ed. Bookman, 2009.
- HOPCROFT, J. F., ULLMAN, J. D., MOTWANI, R. Introdução à Teoria de Autômatos, Linguagens e Computação, Tradução da segunda edição Americana, Elsevier Editora Ltda, 2003.
- PRICE, A. M. A., TOSCANI, S. S., Implementação de Linguagens de Programação: Compiladores. Ed Sagra Luzzatto, 3ª edição, 2008.

- WILHELM, Reinhard; MAURER, Dieter. Compiler Design. Ed. Addison-Wesley, 1995, 606p.
- AHO, A.V.; ULLMAN, J.D. Principles of Compiler Design. Ed Addison-Wesley, 1979, 603p.
- LOUDEN, K. C.. Compiladores - Princípios e Práticas. Editora Thomson, 2004.
- WATT, D., Programming Language Design Concepts, John Wiley and Sons, Ltd, 2004.
- APPLEBY, D., Programming Languages – Paradigm and Practice. Ed McGraw-Hill, Inc., 1991.