

**UNIVERSIDADE FEDERAL DE SANTA CATARINA**  
**CENTRO TECNOLÓGICO**  
**DEPARTAMENTO DE INFORMÁTICA E DE ESTATÍSTICA**  
**PLANO DE ENSINO**

**I. Identificação**

1. **Disciplina:** INE5421 - Linguagens Formais e Compiladores
2. **Turma(s):** 05208
3. **Carga horária:** 72 horas-aula    Teóricas: 72    Práticas: 0
4. **Período:** 2º semestre de 2013
5. **Curso(s):** Ciências da Computação (208)
6. **Requisito(s):**
  - o INE5408 - Estruturas de Dados
  - o INE5415 - Teoria da Computação

**II. Ementa:** O processo de compilação. Linguagens e suas representações. Gramáticas: definição formal, classificação (Hierarquia de Chomsky), propriedades, problemas de decisão e aplicações. Gramáticas regulares, autômatos finitos, conjuntos regulares e expressões regulares. Gramáticas livres de contexto. Autômatos de pilha. Teoria de Parsing. Análise léxica e sintática.

**III. Objetivo(s):**

**III.1 - Geral:** Conhecer a teoria das linguagens formais visando sua aplicação na especificação de linguagens de programação e na construção de compiladores.

**III.2 - Específicos:**

1. Adquirir uma visão geral do processo de compilação sob o ponto de vista de implementação.
2. Correlacionar a Teoria das Linguagens Formais com a Teoria da Computação e esta com a Ciência da Computação.
3. Adquirir sólidas noções de linguagens formais e suas representações.
4. Ser capaz de especificar linguagens através de autômatos e gramáticas.
5. Conhecer e saber usar as técnicas formais de análise léxica e sintática.

**IV. Conteúdo Programático, Metodologia e Carga Horária**

<b>Conteúdo Programático</b>	<b>Método -logia</b>	<b>Carga Horária</b>
1. Introdução: Compiladores, Teoria da Computação e Teoria das Linguagens Formais.	AEX	8
2. Gramáticas: Motivação. Definição formal. Derivação e redução. Sentença, forma sentencial e linguagens. Tipos de gramáticas. Sentença vazia. Recursividade das G.S.C.	AEX	18
3. Autômatos finitos: Determinísticos (AFD) e Não Determinísticos (AFND). Transformação de AFND para AFD. Relação entre AF e GR. Minimização de AFD. Conjuntos regulares e Expressões Regulares. Implementação de AF. Propriedades e problemas de decisão das L.R. Aplicações de A.F. e E.R.	AEX	16
4. Gramáticas livre de contexto (GLC) e autômatos de pilha (PDA): Introdução. Árvore de derivação. Formas de derivação em GLC. Gramática ambígua. Transformações em GLC. Tipos especiais de GLC. PDA. Equivalência entre PDA e GLC. Propriedades e problemas de decisão das LLC. Aplicações.	AEX	14
5. Análise Sintática: Introdução. Classes de analisadores. Analisadores ascendentes e descendentes: Estudo das principais técnicas.	AEX	16

(AEX=Aula Expositiva; LAB=Aula de laboratório; APR=Aula prática; OTR=Outros).

## V. SISTEMA DE AVALIAÇÃO

**V.1 - Critério para Aprovação:** Será aprovado o aluno que obtiver Média Final (MF) igual ou superior a 6.0 e frequência igual ou superior a 75%. A MF será calculada pela seguinte fórmula:

$$MF = (\text{Prova I} + \text{Prova II} + \text{Prova III} + \text{MTE}) / 4$$

onde:  $MTE = (T1 + T2 + MLE) / 3$  e  $MLE = (LE0 + \dots + LE4) / 5$

LE0 = média dos exercícios de fixação (de 5 a 10) dados de uma aula para a outra

LE1 a LE4 = Listas de exercícios referentes aos itens 2 a 5 do programa

**V.2 - Recuperação:** Se MF for maior ou igual 3 e menor que 6, o aluno poderá fazer uma prova de recuperação (REC) e a Nota Final (NF) será  $NF = (MF + REC) / 2$ . Caso contrário,  $NF = MF$ .

### V.3 – Cronograma de Avaliações

Tópico Avaliado	Forma	Semana Provável
Itens 1 e 2	Prova I	7 <sup>a</sup>
Itens 3 e 4	Prova II	13 <sup>a</sup>
Itens 4 e 5	Prova III	17 <sup>a</sup>
Itens 1 a 5	Exercícios	variável
Itens 2 a 5	Trabalhos I e II (Programação)	variável
Itens 1 a 5	Prova de Recuperação	18 <sup>a</sup>

## VI. Bibliografia Básica

1. FURTADO, O. J. V. Apostila de Linguagens Formais e Compiladores – versão 2 - UFSC, 2002, disponível em [www.inf.ufsc.br/~olinto](http://www.inf.ufsc.br/~olinto)

## VII. Bibliografia Complementar

1. RAMOS, M. V. M., NETO, J. J., VEJA, I. S., Linguagens Formais: Teoria, Modelagem e Implementação. Ed. Bookman, 2009.
2. 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.
3. AHO, A. V., SETHI, R., ULLMAN, J. D.. Compiladores – Princípios, Técnicas e Ferramentas, LTC – Livros Técnicos e Científicos Editora S. A., 1995 / Ed. Addison Wesley 2008.
4. FISHER, C. N., CYTRON, R. K., LeBLANC Jr., R. J.. Crafting a Compiler. Ed. Addison Wesley, 2010.
5. MENESES, P. B. Linguagens Formais e Autômatos, Ed. Sagra Luzzato, 5. edição, 2005.
6. SIPSER, Michel. Introdução à Teoria da Computação. Ed. Thomson (Tradução da segunda edição norte-americana), 2007.
7. HOPCROFT, J. E., ULLMAN, J. D. Formal Languages and Their Relations to Automata. Addison-Wesley, 1969.
8. HOPCROFT, J. F., ULLMAN, J. D. Introduction to Automata Theory, Languages and Computation. Ed. Addison-Wesley, 1979.
9. SUDKAMP, T. A. Languages and Machines – An Introduction to the Theory of Computer Science, 2. edição, Ed. Addison Wesley, 1997.
10. LEWIS, H. R. e PAPADIMITRIOU, C. H. , Elementos de Teoria da Computação, Ed. Bookman, 2. edição, 1998.
11. DIVERIO, T. A., MENEZES, P. B., Teoria da Computação – Máquinas Universais e Computabilidade. Ed. Sagra Luzzatto, Porto Alegre, 1999.
12. WOOD, D. Theory of Computation. Ed. John Wiley & Sons, 1987.