



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**  
**FACULDADE DE COMPUTAÇÃO**  
**COLEGIADO DO CURSO DE SISTEMAS DE INFORMAÇÃO**

**FICHA DE DISCIPLINA**

**DISCIPLINA: COMPILADORES**

|                                |                        |   |                          |                  |
|--------------------------------|------------------------|---|--------------------------|------------------|
| <b>CÓDIGO: GSI044</b>          |                        | <b>UNIDADE ACADÊMICA:</b><br><b>FACULDADE DE COMPUTAÇÃO</b> |                          |                  |
| <b>PERÍODO/SÉRIE: 7º OU 8º</b> |                        | <b>CH TOTAL TEÓRICA:</b>                                    | <b>CH TOTAL PRÁTICA:</b> | <b>CH TOTAL:</b> |
| <b>OBRIGATÓRIA: ( )</b>        | <b>OPTATIVA: ( X )</b> | 60  | 00                       | 60               |
| <b>OBS:</b>                    |                        |   |                          |                  |
| <b>PRÉ-REQUISITOS: NÃO HÁ</b>  |                        | <b>CÓ-REQUISITOS: NÃO HÁ</b>                                |                          |                  |

**OBJETIVOS**

Ao final do curso, o aluno deverá ser capaz de definir gramáticas concretas e implementar uma linguagem de programação simples contendo declarações de variáveis globais e de procedimentos e funções com contexto local usando as técnicas estabelecidas de construção de compiladores.

**EMENTA**

Gramáticas. Autômatos. Análise léxica. Análise sintática. Geração de código. Recuperação de erros. Meta-compiladores.

**DESCRIÇÃO DO PROGRAMA**

1. Visão Geral do Curso
2. Gramáticas Regulares, Expressões Regulares e Análise Léxica
3. Gramáticas Livres de Contexto e Análise Sintática
  - a. Transformações em gramáticas
  - b. Algumas propriedades importantes de gramáticas: First e Follow
  - c. Análise Top-Down
    - i. Método LL(1)

- ii.com Descida Recursiva
- d. Análise Bottom-UP (ou Shift-Reduce)
  - i. Método LR(0)
  - ii.Método SLR(1)
  - iii. Método LR(1)
  - iv. Método LALR(1)
- e. Tratamento de erros sintáticos
- 4. Tradução dirigida por sintaxe e Análise Semântica
  - a. Construção de árvore de sintaxe abstrata
  - b. Construção da tabela de símbolos
  - c. Verificação de tipos
- 5. Ambientes de Execução e Geração de código
  - a. Pilha de execução
  - b. Máquinas de execução
    - i. Máquinas de registradores
    - ii.Máquinas de pilha
      - 1. Máquina TAM
  - c. Geração de código intermediário
  - d. Geração de código
- 6. Tópicos Especiais e Revisões
  - a. Geradores de analisadores/tradutores dirigidos por sintaxe
    - i. JFlex / Cup (Lex / Yacc)
  - b. Alocação de registradores
  - c. Otimização de código

## BIBLIOGRAFIA

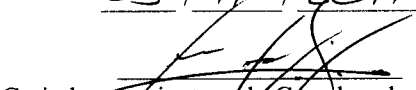
### Básica

AHO, A., et al. Compiladores: Princípios, Técnicas e Ferramentas. Pearson – Addison-Wesley. 2007. 2ª. ed.  
LOUDEN, K. Compiladores: Princípios e Práticas. Thomson Pioneira. 2004.  
GRUNE, D., et al. Projeto Moderno de Compiladores: implementação e aplicações. Campus, 2001.

### Complementar

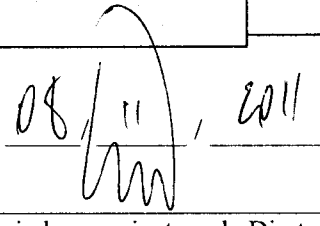
APPEL, A. W. Modern Compiler Implementation in Java, Cambridge University Press, 2002. 2ª.ed.  
WATT, D., BROWN, D. Programming Language Processors in Java: compilers and interpreters. Prentice Hall. 2000.  
MÁRCIO DELAMARO. Como construir um compilador utilizando ferramentas Java. Novatec. 2004.

## APROVAÇÃO

03/11/2011  
  
Carimbo e assinatura do Coordenador do curso

Universidade Federal de Uberlândia  
Prof. Dr. Daniel Gomes Mesquita

Coordenador do Curso de Sistemas de Informação  
Portaria R Nº. 1257/10

08/11/2011  
  
Carimbo e assinatura do Diretor da  
Unidade Acadêmica

Universidade Federal de Uberlândia

Prof. Ilmério Reis da Silva  
Diretor da Faculdade de Computação  
Portaria R Nº. 757/11