

PLANO DE ENSINO

1. IDENTIFICAÇÃO

Curso: Ciência da Computação

Componente Curricular: Linguagens Formais e Autômatos

Fase: 5 (cinco)

Ano/Semestre: 2012/1

Numero de Créditos: 4 (quatro)

Carga horária - Hora Aula: 72 horas

Carga horária - Hora Relógio: 60 horas

Professor: Braulio Adriano de Mello

2. Objetivo Geral do Curso

O curso tem por objetivo a formação integral de novos cientistas e profissionais da computação, os quais deverão possuir conhecimentos técnicos e científicos e serem capazes de aplicar estes conhecimentos, de forma inovadora e transformadora, nas diferentes áreas de conhecimento da Computação. Adicionalmente, os egressos do curso deverão ser capazes de adaptar-se às constantes mudanças tecnológicas e sociais, e ter uma formação ao mesmo tempo cidadã, interdisciplinar e profissional

3. EMENTA

Alfabeto, linguagens, gramáticas e suas representações. Gramáticas: definição formal, classificação (Hierarquia de Chomsky), propriedades. Gramáticas regulares, autômatos finitos, conjuntos regulares e expressões regulares. Gramáticas livres de contexto. Autômatos de pilha. Linguagens sensíveis ao contexto e irrestritas. Análise léxica e sintática.

4. JUSTIFICATIVA

A disciplina de Linguagens Formais e Autômatos discute fundamentos de teoria de linguagens a partir dos conceitos de computabilidade. A compreensão de teoria de linguagens é requisito para a matéria de compiladores e importante para formação em computação em matérias que utilizam ou aplicam fundamentos da hierarquia de linguagens.

5. OBJETIVOS

5.1. GERAL:

Compreender e aplicar os fundamentos de teoria de linguagens na forma de máquinas geradoras e reconhecedoras.

5.2. ESPECÍFICOS:

Domínio dos fundamentos de hierarquia de linguagens

Compreensão dos teoremas para construção de máquinas formais

Identificação e uso de formalismos e teoremas conforme classificação de linguagens

Compreensão e aplicação dos métodos de análise de linguagens

6. CRONOGRAMA E CONTEÚDO PROGRAMÁTICO

Horas	Número Encontro	Conteúdo
2	1	Apresentação da disciplina, plano de aula, sistema de avaliação
3	2	Conceitos iniciais: Autômatos, Computabilidade (Church), decidibilidade (paradoxo de Russel), noções de compiladores
2	3	Teoria de linguagens: Símbolo, alfabeto, cadeia, sentença, concatenação, fechamento, sufixo, prefixo
3	4	Definição formal de gramática gerativa e derivação Tipos de gramáticas, construção de gramáticas
2	5	Construção de gramáticas e regras de derivação
3	6	Linguagens definidas por gramáticas, hierarquia de linguagens (Chomsky), máquinas reconhecedoras
2	7	Construção de gramáticas (exercícios)
3	8	Construção de gramáticas (exercícios)
2	9	Gramáticas Regulares / diagrama de transição / tabela de transição
3	10	Gramáticas Livres de Contexto
2	11	Gramáticas Sensíveis ao Contexto e Irrestritas (Máquina de Turing como máquina reconhecedora de linguagens definidas por GSC e gramáticas irrestritas). Linguagens Enumeráveis Recursivamente e Recursivas.
3	12	Linguagens regulares, expressões regulares
2	13	Construção de expressões regulares
3	14	Autômato Finito (AFD e AFND), Teorema de determinização (Trabalho 1 : implementar o teorema de determinização. Entrada: Gramática regular não determinística. Saída: Autômato finito determinístico)
2	15	Eliminação de épsilon transições em AFDs
3	16	Revisão
2	17	Avaliação 1
3	18	Minimização de AFDs
2	19	Processo para a construção de analisadores léxicos
0	01/05/12	(dia do trabalho conforme calendário)
2	20	Apresentação do Trabalho 1
3	21	Linguagens Livres de Contexto e Árvores de derivação (analisadores sintáticos)
2	22	Máquina reconhecedora (Autômato de Pilha), reconhecedores ascendentes e descendentes
3	23	Teoremas de simplificação de GLCs: Eliminação de símbolos improdutivos.
2	24	Teoremas de simplificação de GLCs: Eliminação épsilon produções (Trabalho 2 : Implementar o teorema. Entrada: GLC com épsilon produções. Saída: GLC livre de épsilon produções)
3	25	Teoremas de simplificação de GLCs: Eliminação de produções unitárias
2	26	Teoremas de simplificação de GLCs: Eliminação de recursão a esquerda e Fatoração
3	27	Teoremas de simplificação de GLCs: Construção dos conjuntos First e Follow
3	28	Avaliação 2
2	29	Apresentação do Trabalho 2
72		
	02/07/12	Prova substitutiva – recuperação de rendimento (todo o conteúdo visto na disciplina).

Obs.: o cronograma/conteúdo pode sofrer modificações

7. PROCEDIMENTOS METODOLÓGICOS (estratégias de ensino, equipamentos, entre outros)

Conduzir a disciplina com aulas expositivas/dialogadas enquanto discutidos os itens de cunho teórico, evoluindo em tópicos específicos para exercícios práticos, demonstrações, contextualização. Atividades práticas em implementação e experimentação.

8. AVALIAÇÃO DO PROCESSO ENSINO-APRENDIZAGEM

Uso de abordagens tais como: provas teóricas, avaliação escrita em aula, exercícios extra-classe, trabalhos de implementação, elaboração de texto/artigo, seminários entre outros trabalhos com complexidade variada.

Avaliação de trabalhos: 50% da nota pela parte escrita (grupo) e 50% da nota em avaliação exposição (individual).

Trabalho facultativo: Pode ser realizado apenas para recuperação de nota. O acréscimo de nota é limitado a 3 pontos num total de 100. Individual. Pode ser apresentado até o último dia letivo de aula.

Avaliações da disciplina:

NP1: uma prova teórica (0,8) e um trabalho de implementação (0,2)

NP2: uma prova teórica (0,8) e um trabalho de implementação (0,2)

Prova de recuperação de rendimento (substituir uma das 2 notas de prova teórica)

Horário de atendimento dos estudantes: Terça-feira; 10:10 as 11:30

9. REFERÊNCIAS

9.1. BÁSICAS:

HOPCROFT, J. F., ULLMAN, J. D., Motwani, R., “Introdução à teoria dos autômatos”, Ed. Campus, 2002.

MENESES, P. B., “Linguagens Formais e Autômatos”, Ed. Sagra Luzzato, 2. edição, 1998.

AHO, A. V., SETHI, R., LAM, M., “Compiladores: Princípios, técnicas e ferramentas”, Ed. Longman do Brasil, 2007.

9.2. COMPLEMENTARES:

PRICE, Ana M. Alencar e Toscani, Simão Sirineo. Implementação de Linguagens de Programação: Compiladores. Bookman Companhia Ed., 2008.

WOOD, D. , “Theory of Computation”, Ed. John Wiley & Sons, 1987.

SUDKAMP, T. A., “Languages and Machines – An Introduction to the Theory of Computer Science, 2. edição”, Ed. Addison Wesley, 1997.

LEWIS, H. R. e PAPADIMITRIOU, C. H. , “Elementos de Teoria da Computação”, Ed. Bookman, 2. edição, 1998.

SIPSER, Michael. Introdução à Teoria da Computação. 2a ed., Thomson, 2007.

DIVERIO, T. A., MENEZES, P. B., “Teoria da Computação – Máquinas Universais e Computabilidade”, Ed. Sagra Luzzato, Porto Alegre, 1999.