

Instituto de Informática

Departamento de Informática Aplicada

Dados de identificação

Disciplina: **COMPILADORES**

Período Letivo: **2015/2**

Período de Início de Validade : **2014/1**

Professor Responsável: **LUCAS MELLO SCHNORR**

Sigla: **INF01147**

Créditos: 4

Carga Horária: 60h

Súmula

Análise léxica e sintática. Tradução dirigida por sintaxe. Otimização de código. Processadores de linguagens.

Currículos

Currículos	Etapa Aconselhada	Pré-Requisitos	Natureza
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO	6	(INF05516) SEMÂNTICA FORMAL N	Obrigatória
ENGENHARIA DE COMPUTAÇÃO	6	(INF05501) TEORIA DA COMPUTAÇÃO N	Obrigatória

Objetivos

O objetivo da disciplina é que os alunos entendam como funciona um compilador, tanto em nível teórico (algoritmos e estruturas de dados envolvidos no projeto de um compilador) como em nível prático. Devem saber projetar e implementar um pequeno compilador para uma linguagem de programação simples, tipicamente um sub-conjunto de uma linguagem imperativa de tipo Pascal ou C.

Conteúdo Programático

Semana	Título	Conteúdo
1	Apresentação da disciplina. Introdução a Compiladores.	Apresenta-se a disciplina e faz-se uma introdução à mesma.
2	Análise léxica - expressões regulares e reconhecedores.	Trata-se da primeira fase do compilador, a análise léxica e das linguagens regulares usadas para reconhecer as estruturas necessárias nessa fase.
2 a 5	Análise sintática	Gramáticas livres de contexto, reconhecedores : parsers top-down e bottom-ups ; conjuntos First e Follows ; tabelas preditivas e LR.
6	Análise semântica	Atributos semânticos herdados e sintetizados ; esquemas S e L atribuídos
7 a 11	Geração de código intermediário.	Caso da atribuição, da checagem de tipos, dos vetores e das estruturas de controle de fluxo de execução (laços, testes).
12 a 13	Suporte ao run-time.	Registro de ativação - transferência de parâmetros.
13 a 14	Assembly	geração de código Assembly e otimizações.
14 a 15	Otimizações	Otimizações genéricas; grafo de fluxo. Otimizações em laços.

Metodologia

As 60 horas previstas para atividades teóricas e práticas indicadas neste Plano de Ensino incluem 30 encontros de 100 minutos de duração (2 períodos de 50 minutos por encontro, 2 encontros por semana, durante 15 semanas), num total de 3.000 minutos, e mais 10 horas (600 minutos) de atividades autônomas, realizadas sem contato direto com o professor, correspondentes a exercícios e trabalhos extraclasse, conforme Resolução 11/2013 do CEPE/UFRGS, Artigos 36 a 38. O Professor poderá se valer de aulas presenciais ou à distância (utilização de recursos da EAD), assim como do apoio de Professores Assistentes (Alunos de Pós-Graduação) em Atividades Didáticas.

Em cada encontro, os alunos poderão fazer perguntas, interagir e consultar o material disponibilizado on-line pelo professor, além de consultar as referências bibliográficas disponíveis na biblioteca. Trabalhos e provas serão aplicados para verificar o entendimento dos conceitos e métodos vistos durante a disciplina.

Carga Horária

Teórica: 60 horas

Prática: 0 horas

Experiências de Aprendizagem

Opcionalmente, os alunos podem:

- programar um analisador léxico, por exemplo com a ferramenta tradicional Lex, ou pela implementação de qualquer outro algoritmo.
- programar um analisador sintático, por exemplo com a ferramenta tradicional Yacc, ou pela implementação de qualquer outro algoritmo.
- programar ações semânticas para gerar código, simplificado ou não.
- implementar um conversor código abstrato - assembly.

Critérios de Avaliação

Provas Escritas

Será realizada pelo menos uma prova teórica P, num dia previamente informado, envolvendo todo o conteúdo das aulas anteriores à prova. O formato da prova inclui perguntas e respostas escritas e opcionalmente exercícios com o uso do computador. Não será aceito aluno na prova a partir de 15 minutos após o início da mesma. Na eventual aplicação de mais de uma prova ao longo do semestre, as datas e os pesos de cada uma devem ser divulgadas pelo professor juntamente com o cronograma. Independente da quantidade de provas aplicadas ao longo do semestre, elas têm peso 1 sobre o total da nota da disciplina.

Trabalhos Práticos

Os trabalhos deverão ser realizados no horário das aulas e fora delas e entregues na forma eletrônica (upload no Moodle) nas datas indicadas. Haverá encontros com um tutor (professor, monitor, ou aluno de pós-graduação em atividade didática) para apresentação de cada um dos trabalhos. A soma T das notas dos trabalhos tem peso 1 sobre o total da nota da disciplina.

Ressalta-se que qualquer tentativa de copiar partes dos programas a serem entregues, ou de obtê-las através de práticas contrárias às

régas do código disciplinar discente, resultará imediatamente na nota zero na disciplina. (Cita-se aqui o artigo 9 do Código Disciplinar Discente (RESOLUÇÃO Nº 07/2004 do CEPE): "Art. 9º – São infrações disciplinares discentes graves: X – apresentar, em nome próprio, trabalho que não seja de sua autoria;").

Formação do Conceito Final

A média ponderada das provas e trabalhos (P+T)/2 será convertida em conceito através da tabela abaixo, levando-se também em conta nesse conceito a participação em aula, interesse, assiduidade e outros critérios subjetivos.

Nota Conceito

>= 9,0 A
>= 7,5 e < 9,0 B
>= 6,0 e < 7,5 C
< 6,0 D

Atividades de Recuperação Previstas

Uma prova teórica poderá ser recuperada numa data única, combinada com o professor no fim do semestre, APENAS no caso de o aluno ter obtido pelo menos a nota 6 nos trabalhos (T >= 6).

Cada nota parcial compondo a nota T poderá ser recuperada até o valor máximo de 80% do valor original, pela entrega de uma nova versão do trabalho corrigindo os eventuais defeitos, num prazo estipulado pelo professor.

Recuperação por falta justificada

No caso de falta justificada por motivo de saúde à prova teórica ou a um encontro de apresentação dos trabalhos, o aluno poderá recuperá-la em data, horário e local a serem marcados pelo professor.

Bibliografia

Básica Essencial

Aho, Alfred V.; Sethi, Ravi; Ullman, Jeffrey D. - Compiladores :princípios, técnicas e ferramentas - Editora LTC (ISBN: 8521610572)

Grune, Dick - Projeto moderno de compiladores :implementação e aplicações - Editora Campus (ISBN: 8535208763)

Levine, John R.; Mason, Tony; Brown, D. - Lex - Editora O'Reilly (ISBN: 9781565920002)

Básica

Keith D. Cooper e Linda Torczon - Engineering a Compiler - Editora Elsevier (ISBN: 978-0-12-088478-0)

Complementar

Kernighan, Brian W.; Ritchie, Dennis M. - The C programming language - Editora Prentice Hall (ISBN: 0131103628)

Outras Referências

Título	Texto
The Lex & Yacc Page	http://dinosaur.compilertools.net/

Observações

Nenhuma observação incluída.