



UNIVERSIDADE FEDERAL DE SANTA CATARINA
Campus Araranguá
Centro de Ciências, Tecnologias e Saúde
Departamento de Computação
PROGRAMA DE ENSINO

I. IDENTIFICAÇÃO DA DISCIPLINA

CÓDIGO	NOME DA DISCIPLINA	Nº DE HORAS-AULA SEMANAIS		TOTAL DE HORAS-AULA SEMESTRAIS	MODALIDADE
		TEÓRICAS	PRÁTICAS		
DEC7565	CONSTRUÇÃO DE COMPILADORES	2	2	72	Presencial

II. PRÉ-REQUISITO(S)

CÓDIGO	NOME DA DISCIPLINA	CURSO
--------	--------------------	-------

III. CURSO(S) PARA O(S) QUAL(IS) A DISCIPLINA É OFERECIDA

Graduação em Engenharia de Computação

IV. EMENTA

Linguagens-fonte, objeto, de alto-nível e de baixo-nível. Especificação de linguagens de programação. Compilação e interpretação. Processadores de linguagens de programação. Máquinas reais e virtuais. Bootstrapping. Análise sintática. Análise de contexto. Ambientes de execução. Geração de código. 90 Otimização de código independente de máquina. Otimização de código dependente de máquina.

V. OBJETIVOS

Objetivo Geral:

Capacitar o aluno na síntese, análise e manipulação de especificações de linguagens de programação de alto nível, assim como no emprego de técnicas de implementação de processadores de linguagens.

Objetivos Específicos:

- Estudar e conhecer os princípios de um compilador;
- Estudar o processo de análise léxica e semântica em um compilador;
- Estudar o processo de geração de código intermediário e código objeto final;
- Estudar o processo de otimização de código intermediário e código objeto final.

VI. CONTEÚDO PROGRAMÁTICO

Conteúdo Teórico seguido de Conteúdo Prático com desenvolvimento de problemas em computador:

Unidade 1: Introdução

Apresentação da disciplina (ementa, bibliografia, metodologia e avaliações)

Introdução aos compiladores

Fases de um compilador

Unidade 2: Análise Léxica

Expressão regular

Reconhecedores

Autômatos finitos para análise léxica

Unidade 3: Análise Sintática

Gramáticas livres de contexto

Análise sintática top-down e bottom-up

Conjuntos FIRST e FOLLOW

Analizador sintático LR

Reconhecedores

Unidade 4: Análise Semântica

Atributos semânticos herdados e sintetizados

Esquemas S e L atributos

Unidade 5: Geração e Otimização de Código

Geração de código intermediário

Otimização de código intermediário

Geração de código objeto
Otimização de código objeto

VII. BIBLIOGRAFIA BÁSICA

1. AHO, Alfred V. et al. Compiladores: Princípios, Técnicas e Ferramentas. 2a ed. São Paulo: Pearson Addison Wesley, 2008.
2. PRICE, Ana Maria de A.; TOSCANI, Simão S. Implementação de Linguagens de Programação: Compiladores. 3a ed. Porto Alegre. Bookman, 2008.
3. ASCENCIO, Ana Fernanda Gomes; CAMPOS, Edilene Aparecida Veneruchi de. Fundamentos da programação de computadores: algoritmos, Pascal, C/C++ e Java. 2. ed. São Paulo: Pearson Prentice Hall, 2008.

VIII. BIBLIOGRAFIA COMPLEMENTAR

1. SEBESTA, Robert W. Conceitos de linguagens de programação. 9. ed. Porto Alegre: Bookman, 2011.
2. PATTERSON, David A.; HENNESSY, John L. Organização e projeto de computadores: a interface hardware/software. 3. ed. Rio de Janeiro: Elsevier, 2005.
3. SANTOS, Rafael. Introdução à programação orientada a objetos usando JAVA. Rio de Janeiro: Campus, 2003
4. MANZANO, José Augusto N. G; OLIVEIRA, Jayr Figueiredo de. Algoritmos: lógica para desenvolvimento de programação de computadores. 27. ed. rev. São Paulo: Érica, 2014. 328 p. ISBN 9788536502212.
5. MEDINA, Marco; FERTING, Cristina. Algoritmos e programação: teoria e prática. 2.ed. São Paulo: Novatec, 2006. 384 p. ISBN 857522073X (broch.).

Os livros acima citados encontram-se na Biblioteca Central e na Biblioteca Setorial de Aranguá (www.bu.ufsc.br).

Aprovação:

O referido programa de ensino foi aprovado na 29^a reunião ordinária do Colegiado do Departamento de Computação em 28 de novembro de 2018.