



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CAMPUS ARARANGUÁ - ARA

PLANO DE ENSINO

SEMESTRE 2017.1

I. IDENTIFICAÇÃO DA DISCIPLINA:

CÓDIGO	NOME DA DISCIPLINA	Nº DE HORAS-AULA SEMANAIS		TOTAL DE HORAS-AULA SEMESTRAIS
		TEÓRICAS	PRÁTICAS	
ARA7565	Construção de Compiladores	2	2	72

HORÁRIO

TURMAS TEÓRICAS

TURMAS PRÁTICAS

08655 – 3.1420-2 e 5-1420-2

MODALIDADE

Presencial

II. PROFESSOR(ES) MINISTRANTE(S)

Prof Gustavo Mello Machado

E-mail: g.mello@ufsc.br

III. PRÉ-REQUISITO(S)

CÓDIGO	NOME DA DISCIPLINA
-	Esta disciplina não possui pré-requisitos

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

Bacharelado em Engenharia de Computação

V. JUSTIFICATIVA

Um bacharel em Engenharia de Computação que queira trabalhar com programação, certamente usa um compilador (ou um interpretador) como ferramenta para traduzir um programa escrito em uma linguagem de alto nível para um programa escrito em linguagem de máquina. Assim, o estudo da construção de compiladores é fundamental para os engenheiros de computadores, pois o domínio dessa ferramenta leva ao desenvolvimento de programas portáteis de alto desempenho e qualidade.

VI. 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. Otimização de código independente de máquina. Otimização de código dependente de máquina.

VII. 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.

prazo de 3 dias úteis apresentando comprovação. (Ver formulário)

XI. CRONOGRAMA PRÁTICO

AULA (semana)	DATA	ASSUNTO
1	06/03/17 a 11/03/17	Unidade 1
2	13/03/17 a 18/03/17	Unidade 2
3	20/03/17 a 25/03/17	Unidade 2
4	27/03/17 a 01/04/17	Unidade 2
5	03/04/17 a 08/04/17	Unidade 2
6	10/04/17 a 15/04/17	Primeira avaliação
7	17/04/17 a 22/04/17	Unidade 3
8	24/04/17 a 29/04/17	Unidade 3
9	01/05/17 a 06/05/17	Unidade 3
10	08/05/17 a 13/05/17	Unidade 3
11	15/05/17 a 20/05/17	Unidade 4
12	22/05/17 a 27/05/17	Unidade 4
13	29/05/17 a 03/06/17	Segunda avaliação
14	05/06/17 a 10/06/17	Unidade 5
15	12/06/17 a 17/06/17	Unidade 5
16	19/06/17 a 24/06/17	Unidade 5
17	26/06/17 a 01/07/17	Terceira avaliação
18	03/07/17 a 08/07/17	Prova de recuperação e divulgação das notas

Obs: O calendário está sujeito a pequenos ajustes de acordo com as necessidades das atividades desenvolvidas.

XII. FERIADOS PREVISTOS PARA O SEMESTRE 2017.1:

DATA	
03/04/2017	Feriado: Aniversário da Cidade
14/04/2017	Feriado: Sexta-Feira Santa
15/04/2017	Dia não letivo
21/04/2017	Feriado: Tiradentes
22/04/2017	Dia não letivo
01/05/2017	Feriado: Dia do Trabalhador
04/05/2017	Feriado: Dia da Padroeira da Cidade
15/06/2017	Feriado: Corpus Christi

XIII. 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.

XIV. BIBLIOGRAFIA COMPLEMENTAR: