



ANEXO II DA RESOLUÇÃO CEPE/UFRR Nº 012, de 18 de agosto de 2020.

PLANO DE ENSINO PARA O ERE

NOME DO CURSO			
Ciência Da Computação			
NATUREZA DO CURSO	(X) Bacharelado () Licenciatura () Tecnológico		
NOME DO PROFESSOR			
Luciano Ferreira Silva			
CÓDIGO / NOME DA DISCIPLINA			
DCC605 - Construção de compiladores			
CATEGORIA	(X) Obrigatória () Eletiva () Optativa Livre () Outro:		SEMESTRE
FORMA DE IMPLEMENTAÇÃO	(X) Regular () Modular por adaptação () Parcial por segmentação		2020.1
CARGA HORÁRIA			DISCIPLINAS PRÉ-REQUISITO(S)
Teórica	Prática	Total	Fundamentos da Computação (DCC 508) Programação em Baixo Nível (DCC 510)
45h	15h	60h	
EMENTA			
Processadores de linguagem; Representações de linguagens; Análise léxica; Análise sintática; Análise semântica; Geração de código.			
OBJETIVOS DE APRENDIZAGEM			
Transmitir os conhecimentos básicos de Linguagens formais e autômatos e Compiladores nos domínios da análise e da aplicação, a fim de classificar os diferentes tipos de linguagens, e conhecer os mecanismos geradores e reconhecedores para cada tipo. Pretende-se ainda, fornecer subsídios ao aluno para implementar o compilador de uma linguagem de programação, desde a definição da linguagem até a construção dos analisadores léxico, sintático e semântico; dos geradores de código intermediário e código de máquina; e dos otimizadores de código; no decorrer do curso e na vida profissional.			
PROGRAMA DA DISCIPLINA			
<ul style="list-style-type: none">PROCESSADORES DE LINGUAGEM: Interpretadores; CompiladoresREPRESENTAÇÕES DE LINGUAGENS: Notação de conjunto; Linguagens; Gramáticas (Produções; Derivações); Classificação de gramáticas; Notações alternativas (Expressões regulares; BNF; Diagramas sintáticos)ANÁLISE LÉXICA: Varredura de tokens; Classificação de tokens (Autômatos finitos; Construção dos autômatos finitos) Analisadores léxicos (Visão conceitual; Aspectos de implementação)ANÁLISE SINTÁTICA: Reconhecimento de sentenças; Derivações canônicas; Árvores sintáticas; Gramáticas ambíguas; Analisadores sintáticos (Autômato de pilha; Analisador sintático preditivo; Analisador de precedência fraca)ANÁLISE SEMÂNTICA: Tabela de símbolos; Heurística para a análise semânticaGERAÇÃO DE CÓDIGO: Geração de código intermediário (Código de três endereços); Otimização de código; Geração de código em linguagem simbólica			
METODOLOGIA DE ENSINO			
Data	Obj. de aprendizagem	Atividades para desenvolver objetivos	Recursos necessários
08/09/2020 à 19/12/2020	Processadores de linguagem Representações de linguagens	Resolução de listas de exercício, participações em fóruns e realização/implementação de trabalhos.	Serão realizados encontros e retidas dúvidas pelos meios: grupo no WhatsApp, SIGAA (com seus recursos); páginas de



UFRR

	Análise léxica		webconferência; e-mails.
	Análise sintática		
	Análise semântica		
	Geração de código.		

AVALIAÇÃO FORMATIVA DA APRENDIZAGEM

Objetivos de aprendizagem	Instrumento avaliativo	Data
Processadores de linguagem; Representações de linguagens; Análise léxica; Análise sintática; Análise semântica; Geração de código.	AC: Avaliação Contínua, realizada gradualmente via resolução de listas de exercícios, participação em fóruns e implementação de trabalhos computacionais. Valor da componente = 10,0 pontos.	08/09/2020 à 11/12/2020
	AF: trabalho prático, no qual será abordada a construção de um compilador completo. A nota será composta por: 50% Aplicação computacional + 30% relatório + 20% apresentação. Valor da componente = 10,0 pontos.	30/11/2020 à 11/12/2020

Cálculo da Nota Final = (AC + AF)/2

Avaliação de Recuperação da Aprendizagem

Objetivos de aprendizagem	Instrumento avaliativo	Data
Processadores de linguagem; Representações de linguagens; Análise léxica; Análise sintática; Análise semântica; Geração de código.	Trabalho de recuperação abordando o conteúdo de menor desempenho do aluno no decorrer do semestre. Valor da componente = 10,0 pontos.	11/12/2020 à 18/12/2020

Cálculo da Nota na Avaliação de Recuperação = Desempenho do aluno na componente

CONDIÇÕES PARA AUXÍLIO QUANDO ESTUDANTES NÃO ATINGIREM OS OBJETIVOS DE APRENDIZAGEM

Aulas de revisão, atividades de pesquisa, exercícios e trabalhos extras, atividades diversificadas que retomem conteúdos não assimilados, trabalhos com técnicas facilitadoras da aprendizagem como anotações, leituras e organização acadêmica.

REFERÊNCIAS RECOMENDADAS

1. BÁSICA

- RICARTE, I. E. Introdução à compilação. Rio de Janeiro: Campus/Elsevier Editora, 2008.
- LOUDEN, K. C. Compiladores: Princípios e Práticas. 2a. ed. São Paulo: Editora Thomson, 2004.
- AHO, A. V.; LAM, M. S. & ULLMAN, R. S. Compiladores: Princípios, técnicos e ferramentas. 2a. ed. São Paulo: Editora Pearson Education, 2008.

2. COMPLEMENTAR

- PRICE, A. M. A. & TOSCANI, S. S. Implementação de linguagens de programação: compiladores. 3a. ed. Porto Alegre: Editora Sagra Luzzatto, 2005.

Professor Responsável
Dr. Luciano Ferreira Silva

Coordenador(a) do Curso