

CTC-41: COMPILADORES

PLANO DE ENSINO 2º SEMESTRE DE 2024

Ementa: Anatomia de um compilador. Análise léxica. Análise sintática: metodologias *top-down* e *bottom-up*. Organização de tabelas de símbolos. Tratamento de erros. Análise semântica. Geração de código intermediário e de código objeto. Meta-compiladores e ferramentas automáticas para construção de compiladores.

Objetivos

Gerais: Apresentar os conceitos fundamentais sobre compiladores através de uma abordagem teórica e prática.

Específicos: Apresentar técnicas consolidadas de projeto e construção de compiladores: fases de análise léxica, sintática e semântica. Uma visão geral do processo de análise e síntese realizado por um compilador. Proporcionar a experiência de projetar e construir um compilador.

Carga horária semanal: 2-0-1-3

Metodologia

- Aulas expositivas com o auxílio de *slides*.
- Aulas práticas sobre as ferramentas utilizadas.

Avaliação

- Duas provas presenciais: Prova 1 (1º bimestre), Prova 2 (2º bimestre).
- Projetos práticos: Projeto 1 (analisador léxico); Projeto 2 (analisador sintático); Projeto 3 (analisador semântico), Projeto final (compilador completo com geração de código)
- Nota do primeiro bimestre: 80% Prova 1, 20% Projeto 1
- Nota do segundo bimestre: 50% Prova 2, 30% Projeto 2, 20% Projeto 3
- Nota do exame: 100% Projeto final
- Provas serão individuais, projetos em grupos de 2 ou 3 alunos

Bibliografia

- AHO, A. V. et al. *Compiladores: princípios, técnicas e ferramentas*. São Paulo: Pearson: Addison-Wesley, 2008.
- LOUDEN, K. C. *Compiladores: princípios e práticas*. São Paulo: Thomson Learning, 2004.
- SANTOS, P. R.; LANGLOIS, T. *Compiladores da teoria à prática*. Rio de Janeiro: LTC, 2018.

Plano de aulas

1º Bimestre

1ª Semana (8/08)

Cap. 1: Introdução

Organização do curso, bibliografia e avaliação
Processo de compilação, hierarquia de Chomsky

2ª Semana (15/08)

Cap. 2: Análise léxica

Expressões regulares (ER)
Autômatos finitos determinísticos (AFD) e não-determinísticos (AFND)
Conversão de ER para AFND

Aula prática

Projetos do curso, formas de entrega, organização dos arquivos

3ª Semana (22/08)

Cap. 2: Análise léxica

Obtenção de um AFD equivalente a um AFND
Minimização de um AFD
A ferramenta *Lex*

4ª Semana (29/08)

Cap. 3: Análise sintática

Notações para gramáticas, derivações, árvores sintáticas, ambiguidades

Aula prática

Tabela de símbolos: estruturas de dados, criação e consulta, uso de *hashing*

5ª Semana (5/09)

Cap. 4: Análise sintática

Análise descendente: preditiva recursiva e não-recursiva

6ª Semana (12/09)

Cap. 4: Análise sintática

Construção da tabela preditiva LL(1)
Recuperação de erros na análise descendente

Aula prática

Dicas sobre Projeto 1

Entrega do Projeto 1 (analisador léxico)

7ª Semana (19/09)

Resolução dos exercícios (Cap. 1 a 4)

8ª Semana (26/09)

Prova 1

Semana de Recuperação

2º Bimestre

1ª Semana (10/10)

Correção da Prova 1

2ª Semana (17/10)

Cap. 5: Análise sintática

Análise ascendente ou redutiva, tabelas SLR(1)

A ferramenta *Yacc*

Aula prática

Uso do *Yacc*, montagem da árvore sintática

3ª Semana (24/10)

Cap. 6: Análise semântica

Verificação de alguns erros de contexto

Linguagem C-: BNF, características, exemplos

4ª Semana (31/10)

Cap. 7: Geração de código

Ambiente de execução, código intermediário

Aula prática

Dicas sobre Projeto 2

Análise semântica com o *Yacc*

Entrega do Projeto 2 (analisador sintático)

5ª Semana (7/11)

Cap. 8: Máquina virtual *Tiny*

Ambiente de execução TM

Funções disponíveis para geração de código

6ª Semana (14/11)

Resolução dos exercícios (Cap. 5 a 8)

Aula prática

Dicas sobre Projeto 3 e Projeto final

Exemplos de geração de código e simulação na TM

7ª Semana (21/11)

(Liberada para finalização dos projetos)

8ª Semana (28/11)

Prova 2

Entrega do Projeto 3 (analisador semântico)

9ª Semana

Exame: Entrega do Projeto final (compilador completo com geração de código)