



UNIVERSIDADE DE SÃO PAULO - ICMC

Departamento de Ciências de Computação

SCC 206 – Introdução a Compilação - 1º Sem /2011

PROFa: Sandra Aluisio

WIKI: [http://wiki.icmc.usp.br/index.php/Scs-206\(sandra\)](http://wiki.icmc.usp.br/index.php/Scs-206(sandra))

I – PROGRAMA

- **Conceitos básicos:** compiladores, interpretadores, montadores, filtros, pré-processadores; estrutura conceitual de um compilador; formas de organização de um compilador; processo de execução de uma linguagem de alto-nível; compilador cruzado (cross-compiler), auto-compilável (bootstrapping), auto-residente e compiler compilers (ou translator writing systems, ou compiler generators).
- **Especificações do Projeto do Compilador Didático: Especificações Gerais do Compilador; Especificações Léxicas, Sintáticas e Semânticas da Linguagem Fonte FRANKIE com as extensões do grupo; Especificações da Linguagem Objeto (MEPA)**
- **Introdução ao compiler compiler JavaCC (<https://javacc.dev.java.net/>)**
- **Análise Léxica:** vantagens da separação entre as análises léxica e sintática; erros Léxicos; especificação e reconhecimento de tokens; formas de implementar a AL; os autômatos finitos e as ER que reconhecem os tokens do programa-fonte.

P1: Gramática da Linguagem Frankie com a extensão do grupo + Analisador Léxico para a Linguagem Estendida

- **Análise Sintática Descendente (ASD ou TOP-DOWN)**
 - com Retrocesso
 - sem Retrocesso (*Predictive Parsers*):
 - Procedimentos Recursivos
 - Dirigida por Tabela (LL(k))
- Recuperação de Erros em ASD com Procedimentos Recursivos

P2: Analisador Sintático para FRANKIE com as extensões do grupo

- **Introdução a Análise Ascendente (ASA ou BOTTOM-UP)**
- **Análise Semântica, Tratamento de Erros Dependentes de Contexto e Checagem de Tipos:** o componente semântico de uma LP; Tarefas da Análise Semântica; Técnicas de Implementação para Tabelas de Símbolos; Ações Semânticas em Compiladores Dirigidos por Sintaxe e Erros da Análise Semântica; Tarefas da Checagem de Tipos; Checagem de tipos Estática versus Dinâmica; Tipo de uma variável ; Compatibilidade de tipos; conversões de tipos; Operadores sobrecarregados.

P3: Analisador Semântico para FRANKIE com extensões e de Rotinas de Checagem Estática de Tipos

- Ambientes de Execução e Geração de Código para uma máquina Hipotética (MEPA): rotinas de suporte ao tempo de execução em várias linguagens (Fortran, C, Pascal); *bindings*; memória em tempo de execução; apresentação da MEPA e suas instruções.

P4: Gerador de Código para a máquina MEPA

II – BIBLIOGRAFIA

- Aho, A. V. et alli - Compilers: Principles, Techniques and Tools. Addison-Wesley Publishing Company, 1986. (Dragãozinho Vermelho em INGLÊS)
- Aho, Lam, Sethi & Ullman. Compilers: Principles, Techniques, and Tools; 2nd Edition, Addison-Wesley Publishing Company, 2006 . (Dragãozinho Roxo em INGLÊS)
- <http://dragonbook.stanford.edu/>
- Aho, A. V. et alli - Compiladores - Princípios, Técnicas e Ferramentas, 2ª Edição, Ed. Pearson, 2007. (Dragãozinho Roxo em PORTUGUÊS)
- Aho, A.V.; Ullman, J.D.; Sethi, R. (1995). Compiladores: Princípios, Técnicas e Ferramentas. Editora LTC. (Dragãozinho Vermelho em PORTUGUÊS)
- Kowaltowsky, T. - Implementação de Linguagens de Programação, Guanabara Dois, 1983.
- Louden, K.C. (2004). Compiladores: Princípios e Práticas. Editora Thomson Learning.
- Price, A.M.A. e Toscani, S.S. (2001). Implementação de Linguagens de Programação: Compilador. Editora Sagra Luzzatto.

III - CRITÉRIO DE AVALIAÇÃO

PROVAS: Haverá 2 provas cujas datas de realização estão marcadas a seguir. Cada prova vale de 0 a 10. Não há substitutiva.

1^o Prova: 25/4 ----- 2^o Prova: 20/6

PROJETO: Um compilador que traduz programas da linguagem FRANKIE (Pascal com partes da sintaxe de C) em código de um compilador hipotético à pilha chamado MEPA (Máquina de Execução para Pascal). Para a implementação desse projeto será fornecida a sintaxe da linguagem fonte – FRANKIE – em notação EBNF. As características gerais da MEPA e a interpretação das instruções serão vistas durante o curso. As instruções da MEPA encontram-se em Kowaltowsky, T. - Implementação de Linguagens de Programação, Guanabara Dois, 1983.

O projeto será desenvolvido por uma equipe de 3 alunos (se sobraremos alunos faremos grupos de 4 ou de 2). O projeto é dividido em 4 entregas, definidas no cronograma abaixo. Cada uma vale de 0 a 10 e suas correções serão baseadas em critérios explícitos definidos previamente, juntamente com a especificação da entrega.

LISTAS DE EXERCÍCIOS. Haverá várias listas. Espera-se que os alunos resolvam as listas e tirem as dúvidas com o monitor PAE ou professor.

CÁLCULO DA MÉDIA:

MP = Média Aritmética das Provas

MT = Média Ponderada das Notas dos Projetos: 10%, 15%, 35%, 40%

MF = Média Final

Se $MP \geq 5$ **então** $MF = (6MP + 4MT)/10$

senão $MF = (7.5MP + 2.5MT)/10$

RECUPERAÇÃO: Será em 11/7 e só terão direito à recuperação os alunos com $3.0 \leq MF < 5.0$ e frequência \geq a 70%.

Média = $MF + (Rec/2,5)$, se $Rec \geq 7,5$; ou

Média = $\text{Max} \{MF, Rec\}$, se $Rec < 5,0$; ou

Média = 5,0, se $5,0 \leq Rec < 7,5$.

Cronograma

Data	Aula	Tópico	Trabalho	Data Entrega
21/2	1	Apresentação do Curso, da linguagem para o projeto, Conceitos básicos		
28/2	2	Especificações léxicas, sintáticas e semânticas da LF Análise Léxica	P1	
14/3	3	Javacc		
21/3	4	AS Descendente		
28/3	5	AS Descendente	P2	P1
4/4	6	Tratamento de erros sintáticos		
11/4	7	Análise Semântica		
25/4	8	Prova 1		P2
2/5	9	Análise Semântica		
9/5	10	Análise Semântica	P3	
16/5	11	Checagem de Tipos		
23/5	12	Checagem de Tipos		
30/5	13	Ambientes de Execução & Geração Código		
6/6	14	Ambientes de Execução & Geração Código	P4	P3
13/6	15	Ambientes de Execução & Geração Código		
20/6	16	Prova 2		
27/6	17			P4