**CES-41: Compiladores**

**Laboratório 4: Tabela de símbolos e de analisador semântico para uma linguagem de programação utilizando a ferramenta Yacc**

Cássio dos Santos Sousa, Renan Pablo Rodrigues da Cruz

Professor: Fábio Carneiro Mokarzel

06 de outubro de 2015

**1. Introdução**

Já foi possível criar, em laboratórios anteriores, um analisador léxico (com uso da ferramenta *Flex*) e um analisador sintático (com uso da ferramenta *Yacc*) para a Linguagem COMP-ITA 2015.

Para este laboratório, como o título alerta, o objetivo é o de construir uma tabela de símbolos e um analisador semântico para a mesma linguagem com o uso das mesmas ferramentas, de tal forma que o programa resultante seja capaz de imprimir o conteúdo da tabela de símbolos e as devidas mensagens de erros para programas quaisquer utilizados como entrada.

Este laboratório e os seguintes são mais complexos que os anteriores, de tal forma que seu desenvolvimento pode ser feito agora por duplas de alunos, contanto que essa dupla seja mantida até a entrega do último laboratório da disciplina (de um total de seis).

**2. Detalhes do código**

O código escrito para este laboratório feito em *Yacc* foi feito com base no código feito no laboratório anterior, realizando primeiro a análise sintática do mesmo. Se o código não possuir erros de sintaxe, ele é apresentado na tela no formato *pretty printer*, removendo comentários e respeitando tabulações e espaçamentos de cada cadeia presente no código.

Assumiu-se, primeiramente, que todos os códigos utilizados como casos teste neste laboratório não possuem erros de sintaxe, uma vez que o objetivo deste laboratório não é checa-los. Concorrentemente à versão *pretty printer* do código, é feita a análise semântica do mesmo, e serão impressas mensagens de erros semânticos sempre que acontecerem.

Feita a impressão do código, se não ocorreu erro de sintaxe, é impressa a tabela de símbolos encontrados no código.

**2.1. Formato da Tabela de Símbolos**

Manteve-se o formato da Aula 6 e da Prática 3 com respeito ao número limite da tabela hash (23 classes possíveis). Se uma das classes possuir símbolos, é impressa a numeração da classe seguida de cada um dos símbolos, um para cada linha.

Para cada símbolo, é impresso o nome (presente na variável s->cadeia),

Se o símbolo corresponder a um array, é impresso “EH ARRAY”, o número de dimensões e o comprimento de cada uma delas.