

Trabalho 1 - Análise Léxica

Prof. Rogério Aparecido Gonçalves

Sumário

1	Análise Léxica	1
1.1	Implementação	1
1.2	Linguagens de programação para a implementação	2
1.3	Documentação	2
1.4	Avaliação	3
1.5	Entrega e apresentação	3
1.6	Referências	3

1 Análise Léxica

1.1 Implementação

A linguagem de programação que será implementada é a T++.

Para implementar o sistema de varredura da linguagem, é necessário tomar nota nas seguintes classes de **tokens**:

palavras reservadas	símbolos
se	+ soma
então	- subtração
senão	* multiplicação
fim	/ divisão
repita	= igualdade
flutuante	, vírgula
retorna	:= atribuição
até	< menor
leia	> maior
escreve	<= menor-igual
inteiro	>= maior-igual

palavras reservadas	símbolos
	(abre-par
	(fecha-par
	: dois-pontos
	[abre-col
] fecha-col

Ainda pode ser definido:

- **número:** 1 ou mais dígitos que podem ser *inteiro* ou *flutuante* (representação em notação científica ou não);
- **identificador:** começa com uma letra e precede com N letras e números sem limite de tamanho;
- **comentários:** cercados de chaves da seguinte forma: {...}

Obs.: Para a construção da linguagem, utilize palavras reservadas em português-brasil como indicado na tabela.

1.2 Linguagens de programação para a implementação

Para a implementação do compilador, pode ser utilizado qualquer linguagem de programação. É recomendado que seja utilizado uma linguagem que dê suporte à estruturas de dados de alto nível e preferencialmente que exista bibliotecas para a construção da varredura e gramática. Algumas LPs/ferramentas conhecidas são:

- C/C++ - Flex/Bison [3][4]
- Python - PLY (que possui ferramenta léxica e sintática) [5]
- Java - JFlex/Jacc [6][7]

Foi adicionado também um vídeo no moodle da disciplina sobre varredura na linguagem de programação Go. Ao longo da disciplina serão adicionados exemplos, tutoriais e outras informações para auxiliar o aluno na disciplina.

1.3 Documentação

Durante toda a disciplina o aluno criará uma documentação formal da implementação da linguagem. Inicialmente a documentação terá que ter:

- Especificação da linguagem de programação T++;
- Especificação formal dos autômatos para a formação de cada classe de token da linguagem;
- Detalhes da implementação da varredura na LP e ferramenta (e/ou bibliotecas) escolhidas pelo projetista.

1.4 Avaliação

Será avaliado o funcionamento da varredura para a linguagem de programação **T++**.

- Varredura: programa de exemplo **T++** de **entrada** na linha de comando. **Saída** será o conjunto de pares (valor:token).

Para a avaliação inicial será considerado então (obrigatoriamente):

- Utilizar palavras reservadas em português-BR;
- Construção da análise léxica;
- Inserção de comentários (para adicionar explicações futuras no código);
- Levantamento de erros (sugerindo classes de erros). Para isso: Contabilizar linhas (`\n`), colunas e lexema atual (as ferramentas fazem isso);

1.5 Entrega e apresentação

O trabalho será **individual** e deverá ser entregue até o dia 29/08/2016 no moodle da disciplina em um pacote compactado. Deverá ser especificado na entrega o mecanismo de execução da varredura para a realização da correção.

1.6 Referências

- [1] LOUDEN, Kenneth C. Compiladores: princípios e práticas. São Paulo, SP: Thomson, c2004. xiv, 569 p. ISBN 8522104220.
- [2] <http://web.cecs.pdx.edu/~mpj/jacc/>
- [3] <http://flex.sourceforge.net/>
- [4] <http://www.gnu.org/software/bison/>
- [5] <http://www.dabeaz.com/ply/>
- [6] <http://jflex.de/>
- [7] <http://web.cecs.pdx.edu/~mpj/jacc/>