



Trabalho Prático 1 -- Entrega: 4/4

Parte 1: Entregar o Relatório do Projeto de Construção do Compilador para FRANKIE com Extensões do Grupo (IMPRESSO), contendo:

1. Especificações Gerais do Compilador (10 tópicos)
2. Especificações da Linguagem Fonte com as extensões do grupo, parte das Especificações Léxicas
3. Especificações da Linguagem Fonte - Definição Formal da Gramática Estendida

Parte 2: Criar a especificação para o Analisador Léxico de FRANKIE com as extensões do grupo na parte do arquivo de especificação usado pelo JavaCC, dedicada ao reconhecimento dos tokens. O Analisador Léxico deve tabular a saída código/token para FRANKIE com as extensões do grupo.

Um Analisador Léxico faz a interface entre o arquivo que contem o programa e o resto do compilador e tem como funcionalidades:

- separar cada elemento do VT da linguagem (token) e devolver um código apropriado (no JavaCC tal código é o nome simbólico do token)
- eliminar comentários, brancos, \n e tabs;
- reconhecer erros léxicos (ex. símbolo não pertencente ao Vt, constantes mal formadas, comentário aberto e não fechado).

Por exemplo, um analisador léxico da linguagem Pascal:

```
program teste1;
var x: integer;
begin
    x := 3;
    if x > 5 then
    begin
        x := 5
    end else
    begin
        x := 0
    end
end.
```

```
66 Prompt de Comando
C:\javacc-5.0\AnaLex\Analizador_Lexico>java AnaLex < teste1.pas
@ (1,1) PROGRAM - program
@ (1,9) ID - teste1
@ (1,15) SEMICOLON - ;
@ (2,1) VAR - var
@ (2,5) ID - x
@ (2,6) COLON - :
@ (2,8) ID - integer
@ (2,15) SEMICOLON - ;
@ (3,1) BEGIN - begin
@ (4,9) ID - x
@ (4,11) ASSIGNMENT - :=
@ (4,14) NUMBER - 3
@ (4,15) SEMICOLON - ;
@ (5,9) IF - if
@ (5,12) ID - x
@ (5,14) LESS - >
@ (5,16) NUMBER - 5
@ (5,18) THEN - then
@ (6,6) BEGIN - begin
@ (7,17) ID - x
@ (7,19) ASSIGNMENT - :=
@ (7,22) NUMBER - 5
@ (8,9) END - end
@ (8,13) ELSE - else
@ (9,6) BEGIN - begin
@ (10,17) ID - x
@ (10,19) ASSIGNMENT - :=
@ (10,22) NUMBER - 0
@ (11,9) END - end
@ (12,1) END - end
@ (12,4) DOT - .
C:\javacc-5.0\AnaLex\Analizador_Lexico>
```

Parte 3: Customizem as mensagens de erros léxicos. JavaCC gera um arquivo para gerenciar os erros léxicos:

- TokenMgrError.java

e testem o parser/scanner gerado com vários programas corretos e com erros. Estes devem ter as extensões do grupo.

Forma de Entrega:

- **Parte 1:** impressa, entregue na aula
- **Parte 2 e Parte 3:** Arquivo de especificação do JavaCC (código fonte), arquivos de teste e arquivo de saída da análise léxica: e-mail do curso (SCC206.2011.A@gmail.com), até 23:59 h
- Código executável (jar) e instruções de como rodar: e-mail do curso: (SCC206.2011.A@gmail.com), até 23:59 h
- A Parte 2 é dependente da Parte 1, assim, só será corrigida se ela for entregue.