

Enunciado Laboratório 01 de Compiladores.
Entrega 26/03/2019

O trabalho pode ser feito em duplas

Escreva um programa de computador que faça análise léxica do texto de um programa de computador em Pascal para o seguinte subconjunto de tokens abaixo. Você pode criar este programa simulando um autômato finito adaptado para casamento do padrão mais longo com múltiplas alternativas de "estado final" ou utilizar alguma ferramenta para descrição de tokens, tais como JavaCC.

Os tokens que devem ser identificados são:

1. palavras reservadas (1.3.1 e 1.3.3)
2. identificadores (1.4)
3. números inteiros em decimal com ou sem sinal (**não é necessário tratar octal, hexa e binário**) (1.6)
4. números ponto flutuante com ou sem sinal (inclusive notação científica, **mas sem octal, hexa e binário**) (1.6)
5. strings
6. símbolos especiais (simples e duplos) e operadores (1.1)

Para padronização dentre os vários dialetos de Pascal, adotaremos a definição descrita em

<https://www.freepascal.org/docs-html/ref/refch1.html>

sem as palavras reservadas de Object.

São os itens do link: 1.1, 1.3.1, 1.3.3, 1.4, 1.6 e 1.8

A saída deve conter as informações descrevendo um token por linha.

1. Cada um dentre palavras reservadas, símbolos especiais e operadores devem ser descritos por uma linha contendo somente a string que o define;
2. Os demais devem ser descritos por linhas começando com um nome especial, seguido de espaço, seguido da string que define o token, sendo que os nomes especiais são:
 - a. "ID" para identificadores;
 - b. "NUM" para números inteiros;
 - c. "FLOAT_NUM" para números de ponto flutuante;
 - d. "STRING" para strings.

Dica: a versão 5.0 do javacc foi mais fácil de instalar no laboratório do IFNMG em 2017.