Especificação do Trabalho - Parte 02

1 Objetivos do Trabalho

Este trabalho tem por objetivos: criar, desenvolver e implementar um analisador léxico e um analisador sintático para uma linguagem de programação.

O trabalho poderá ser feito individual, em dupla ou trio.

2 Linguagem

A partir do trabalho 01, onde foram definidas as regras, autômatos, expressões regulares para uma linguagem de programação a ser desenvolvida; para este trabalho, as regras deverão ser implementadas de acordo com o que foi definido.

As regras ou produções poderão sofrer alterações durante o seu desenvolvimento.

3 Implementação

Para implementar os analisadores léxico e sintático poderá ser utilizadas as linguagens C, C++, Java ou Python, além das ferramentas:

- FLEX (Analisador Léxico – http://www.gnu.org/software/flex/)
- BISON (Analisador Sintático – http://www.gnu.org/software/bison/)
- LEX (Analisador Léxico – http://dinosaur.compilertools.net/)
- YACC (Analisador Sintático – https://yacc.software.informer.com/0.4/)
- Gold Parser (http://goldparser.org/)
- Regex (http://www.regular-expressions.info/java.html ou http://docs.oracle.com/javase/tutorial/essential/regex/)
- JavaCC (https://javacc.org/)
- JFlex (https://jflex.de/)
- GALS (Gerador de Analisador Léxico e Sintático – http://gals.sourceforge.net/)

Deverão ser identificados na linguagem, os seguintes símbolos:

- Palavras reservadas.
- Identificadores (variáveis).
- Números inteiros, reais, caracteres.
- Operadores aritméticos, lógicos, relacionais.
- Comando de atribuição.

Deverão ser desenvolvidos três programas exemplo utilizando os comandos de entrada e saída; comandos de condicionais e os comandos de repetição. Para as linguagens que não são de programação, deverão ser criados três programas exemplo utilizando as estruturas definidas na linguagem. O arquivo de saída poderá ser gerado de duas maneiras:

1. O código fonte poderá ser traduzido para a linguagem C, C++, Java ou Python e compilado utilizando os respectivos compiladores de acordo com a linguagem escolhida.
2. Poderá ser criado um compilador da linguagem desenvolvida, através do qual o programa será compilado e executado.

No instante da compilação deverá ser gerado um arquivo contendo o o reconhecimento de todos os comandos apresentados nos programas exemplo. Este arquivo deverá conter informações identificando cada um dos comandos do arquivo. Por exemplo, para um tipo de dado inteiro definido na linguagem, deverá aparecer INT PALAVRA RESERVADA ou o símbolo = ATRIBUIÇÃO. Quando for encontrado um comando condicional, por exemplo a estrutura if(a>b), este deverá ser reconhecido como COMANDO CONDICIONAL.

4 Documentação

Deverá ser entregue até o dia 24/07/2024 em formato digital, o código-fonte implementado dos analisadores léxico e sintático, assim como, os exemplos dos programas usando as regras da linguagem desenvolvida. Não esqueça de colocar também o arquivo contendo o reconhecimento e identificação de todos símbolos e tokens encontrados nos programas exemplo.

O código-fonte deverá ter um arquivo README, contendo informações do modo que a compilação e a execução será feita.

5 Apresentação

A apresentação ocorrerá nos dias 25/06/2024 e 02/07/2024. Cada grupo fará uma breve apresentação do trabalho, com duração entre 10-15 minutos. Deverá ser brevemente exposta a documentação do trabalho, destacando as principais dificuldades encontradas no desenvolvimento dos analisadores léxico e sintático e as decisões de implementação. Deverá ser apresentado ainda o funcionamento do programa usando dois exemplos.

Todos os integrantes do grupo devem apresentar o trabalho.

6 Avaliação

O trabalho terá um total de 10 pontos, onde 7 pontos serão referentes à implementação e os demais 3 pontos serão atribuídos à apresentação a ser realizada.

Serão descontados pontos de trabalhos onde o programa não compila, trabalhos iguais e aqueles trabalhos que não estão de acordo com o solicitado.

A nota da implementação será a mesma para todos os integrantes do grupo. A nota da apresentação será individual.