

## 1. Descrição do trabalho:

Como parte da nota da segunda avaliação de Compiladores para o Curso de Ciência da Computação, os alunos deverão desenvolver um tradutor de linguagens. Esse tradutor, baseado na técnica de tradução dirigida pela sintaxe deverá conter as fases descritas no item 3 desse documento e ser desenvolvido em Linguagem C, nas plataformas de sistemas operacionais Windows ou GNU/Linux.

## 2. Objetivo

O objetivo principal dessa atividade é implementar alguns mecanismos de compilação e incentivar o estudo da teoria que abrange as fases de um compilador de linguagens de programação.

## 3. As Fases do Compilador:

Como pode ser ilustrado na figura 01, o trabalho consistirá de duas partes distintas:

- Parte 01 – como fase inicial do compilador, o aluno deverá implementar o analisador léxico, usando como base a gramática definida a seguir. O programa deverá ler um arquivo e realizar as operações de análise.
- Parte 02 – após a realização da análise léxica, implemente um analisador Bottom-UP SLR ou LR(1) ou LALR. O programa deve conter a tabela sintática que será usada para ler o mesmo arquivo de entrada e gerar a respectiva análise sintática.

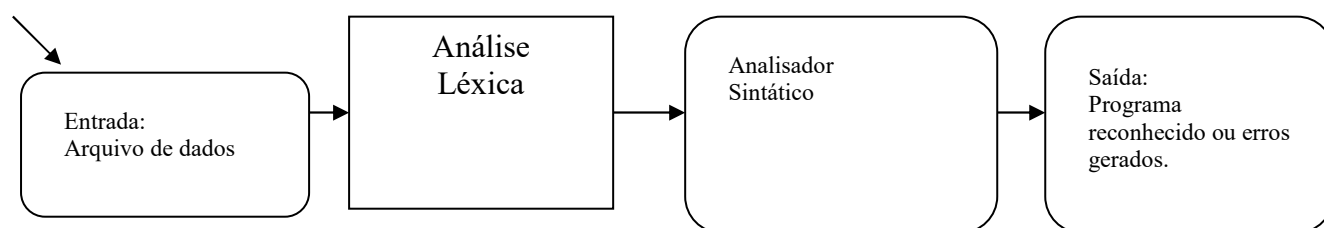


Figura 01 – Fases do Compilador

Gramática:  $G = (V, T, P, S)$

$T = \{ ; \text{ if then else } + * 0 1 ( ) \}$  – tokens

$V = \{ S', S, A, I, E, T, F \}$

$P = \{$

$S' \rightarrow S$

$S \rightarrow E ; A$

$S \rightarrow A$

$A \rightarrow I$

$A \rightarrow E$

$I \rightarrow \text{if } E \text{ then } S \text{ else } S$

$E \rightarrow E + T$

$E \rightarrow F$

$E \rightarrow ( E )$

$T \rightarrow T * F$

$T \rightarrow F$

$F \rightarrow 0$

$F \rightarrow 1$

$\}$

$S'$  = produção inicial

#### 4. Informações sobre Grupos de Trabalho e Data Limite de Entrega:

Esse trabalho terá o valor total de 40% de N2. Os grupos poderão ter até 3 pessoas no máximo, com exceção no caso de faltarem integrantes para cumprir esse requisito.

Obs.: o programa deverá ser VISUAL, com menus de manipulação de arquivos, menu EDITAR com funções de cópia e cola, tela de editor, tela de mensagens de erros após o processo de compilação.

A data final de entrega será dia **22/06/2018**, impreterivelmente.

#### 5. Tratamento de erros e considerações de implementação:

O analisador léxico deverá tratar erros léxicos, parando a compilação sempre que encontrar um erro.

O erro do analisador sintático resume-se somente na questão de qual estado o autômato atinge após ler o arquivo: se for estado final, não emite-se um erro, senão:

"Erro sintatico: Programa invalido.\n"

## 6. Exemplos de Análises válidas:

if ( 0 + 1 ) then 0 + 1 * 1 else 0 + 1 ; 0 + 0 + 0
if ( 0 + 1 ) then 1 + 1 else 1 + 1 ; if 0 then 0 + 1 * 1 * 1 else 1 + 1 ; 1
1 + 1 + 1 ; 1 + 1 * 0