LISTA DE EXERCÍCIOS no3 – ANALISADOR SINTÁTICO

Em cada um dos exercícios a seguir:

1. considere que os *tokens* (palavras reservadas, símbolos especiais e outras categorias definidas no léxico da linguagem) já foram especificados;
2. construa uma gramática, usando a notação BNF, que defina as regras sintáticas para a descrição textual apresentada.

forma geral da declaração de variáveis:

**var**

<lista de identificadores> **:** <tipo> **;**

* <lista de identificadores> **:** <tipo> **;** pode ocorrer uma ou mais vezes.
* em <lista de identificadores> deve existir no mínimo um identificador. Caso existam mais identificadores, os mesmos são separados uns dos outros por vírgula.
* <tipo> pode ser **int**, **float** ou **char**.

forma geral do comando de entrada de dados:

**read (** <lista de identificadores> **)**

* <lista de identificadores> pode conter zero ou mais identificadores, ou seja, a <lista de identificadores> é opcional, mas os parênteses não. Caso existam vários identificadores, os mesmos são separados uns dos outros por vírgula

forma geral do comando de seleção:

**if** <expressão> **then** <lista de comandos> **else** <lista de comandos> **end**

* <lista de comandos> pode conter um ou mais <comando>.
* **else** <lista de comandos>é opcional.

Considere a existência dos não-terminais <expressão> e <comando> que definem, respectivamente, as regras sintáticas de formação das expressões e de todos os outros comandos válidos para a linguagem em questão.

forma geral dos comandos de repetição:

**repeat** <lista de comandos> **until** <expressão>

ou **while** <expressão> **do** <lista de comandos> **end**

* <lista de comandos> pode conter um ou mais <comando>.

Considere a existência dos não-terminais <expressão> e <comando> que definem, respectivamente, as regras sintáticas de formação das expressões e de todos os outros comandos válidos para a linguagem em questão.

forma geral de um programa:

**program** identificador

<declaração de variáveis>

**begin** <lista de comandos> **end** identificador

* <declaração de variáveis>é conforme descrito e já especificado na questão 1.
* <declaração de variáveis>é opcional.
* <lista de comandos> pode conter um ou mais comandos de entrada (conforme descrito e já especificado na questão 2), de seleção (conforme descrito e já especificado na questão 3) e de repetição (conforme descrito e já especificado na questão 4) .

forma geral do comando de seleção:

**case** <expressão> **of**

<opção>**:** <lista de comandos> **end**

**else**

<lista de comandos>

**end**

* <opção> pode ser uma ou mais constantes inteiras e/ou uma ou mais faixas de constantes inteiras (limite inferior e limite superior separados por **..**), sendo que as constantes e as faixas de constantes são separadas umas das outras por vírgula.
* <opção>é seguida por dois pontos (**:**), seguido por uma <lista de comandos> finalizada com **end**.
* <opção> **:** <lista de comandos> **end** pode ocorrer uma ou mais vezes.
* <lista de comandos> pode conter um ou mais <comando>.
* **else** <lista de comandos>é opcional.

Considere a existência dos não-terminais <expressão>e <comando>que definem, respectivamente, as regras sintáticas de formação das expressões e de todos os outros comandos válidos para a linguagem em questão.

1. Considere os diagramas de sintaxe abaixo que especificam a forma geral da definição de uma classe em uma linguagem de programação orientada a objetos:

*definição de classe =*

herança

propriedades

comportamento

*herança = comportamento =*

método

*método = visibilidade =*

parâmetros

*propriedades =*

tipo

visibilidade

Considere também a existência dos não-terminais <tipo>e <parâmetros> que definem, respectivamente, todos os tipos válidos e a forma geral da definição de parâmetros na linguagem em questão. Construa uma gramática, usando a notação BNF, que defina as regras sintáticas para a definição de uma classe conforme o especificado.