Gramática BNF da Linguagem LPN

A gramática BNF (Backus-Naur Form) da linguagem de programação LPN é definida como:

Estrutura do Programa

<programa> ::= <titulo> <inicio> <comando> "\n" <RES> "\n" <fim>

<titulo> ::= "PROGRAMA \"" <ident> "\":" "\n"

<inicio> ::= "INICIO\n"

<fim> ::= "FIM"

Comando RES (Resultado Final)

<RES> ::= "RES" <espaco> "=" <espaco> <expressao>

Comandos e Atribuições

<comando> ::= <atribuicao>

| <comando> "\n" <comando>

<atribuicao> ::= <espaco> <ident> <espaco> "=" <espaco> <expressao>

Expressões e Operadores

<expressao> ::= <expressao1> (<espaco> <operador_soma> <espaco> <expressao1>)*

<expressao1> ::= <expressao2> (<espaco> <operador_mult> <espaco> <expressao2>)*

<expressao2> ::= <numero> | <ident> | "(" <espaco> <expressao> <espaco> ")"

<operador_soma> ::= "+" | "-"

<operador_mult> ::= "*" | "/"

Elementos Básicos

<espaco> ::= (" " | "\t")*

<ident> ::= ([a-z] | [A-Z])+

<numero> ::= <positivo> | <negativo>

<positivo> ::= ("0" | [1-9] [0-9]*) ("," [0-9]+)? <cientifico>?

<negativo> ::= "-" <positivo>

<cientifico> ::= "e" "-"? [1-9] [0-9]*

Exemplos Válidos de Programas LPN

Exemplo 1: Programa Simples

PROGRAMA "SOMA":

INICIO

A = 5

B = 10

RES = A + B

FIM

Exemplo 2: Expressões com Parênteses

```
PROGRAMA "FORMULA":
INICIO
A = 2
B = 3
C = 4
RES = A * (B + C) / (B - A)
FIM
```

Semântica da Linguagem

1. Todas as variáveis são consideradas inteiras.
2. As variáveis não precisam ser declaradas antes de serem usadas.
3. A variável especial `RES` armazena o resultado final do programa.
4. As operações aritméticas seguem a precedência matemática padrão:
   - Parênteses têm a maior precedência
   - Multiplicação e divisão têm precedência sobre adição e subtração
   - Operações de mesma precedência são avaliadas da esquerda para a direita
5. O compilador não verifica divisão por zero.
6. Operações de multiplicação e divisão são implementadas através de repetição de operações de adição e subtração, respectivamente.