**Linguagem LALG**

1. <programa> ::= program ident ; <corpo> .
2. <corpo> ::= <dc> begin <comandos> end
3. <dc> ::= <dc_v> <dc_p>
4. <dc_v> ::= var <variaveis> : <tipo_var> ; <dc_v> | λ
5. <tipo_var> ::= real | integer
6. <variaveis> ::= ident <mais_var>
7. <mais_var> ::= , <variaveis> | λ
8. <dc_p> ::= procedure ident <parametros> ; <corpo_p> <dc_p> | λ
9. <parametros> ::= ( <lista_par> ) | λ
10. <lista_par> ::= <variaveis> : <tipo_var> <mais_par>
11. <mais_par> ::= ; <lista_par> | λ
12. <corpo_p> ::= <dc_loc> begin <comandos> end ;
13. <dc_loc> ::= <dc_v>
14. <lista_arg> ::= ( <argumentos> ) | λ
15. <argumentos> ::= ident <mais_ident>
16. <mais_ident> ::= ; <argumentos> | λ
17. <pfalsa> ::= else <cmd> | λ
18. <comandos> ::= <cmd> ; <comandos> | λ
19. <cmd> ::= read ( <variaveis> ) |
    write ( <variaveis> ) |
    while <condicao> do <cmd> |
    if <condicao> then <cmd> <pfalsa> |
    ident := <expressao> |
    ident <lista_arg> |
    begin <comandos> end
20. <condicao> ::= <expressao> <relacao> <expressao>
21. <relacao> ::= = | <> | >= | <= | > | <
22. <expressao> ::= <termo> <outros_termos>
23. <op_un> ::= + | - | λ
24. <outros_termos> ::= <op_ad> <termo> <outros_termos> | λ
25. <op_ad> ::= + | -
26. <termo> ::= <op_un> <fator> <mais_fatores>
27. <mais_fatores> ::= <op_mul> <fator> <mais_fatores> | λ
28. <op_mul> ::= * | /
29. <fator> ::= ident | numero_int | numero_real | ( <expressao> )

Além disso, inclusão do comando repeat-until.

Comentários entre chaves { }

Identificadores e números são itens léxicos da forma:
- ident: seqüência de letras e dígitos, começando por letra
- número inteiro: seqüência de dígitos (0 a 9)
- número real: pelo menos um dígito seguido de um ponto decimal e de uma seqüência de um ou mais dígitos

**Exemplos de programas LALG**

```
program nome1;
{exemplo 1}
var a, a1, b: integer;
begin
read(a, b);
a1:= a + b;
while a1>a do
begin
write(a1);
a1:= a1-1;
end;
if a<> b then write(a);
end.
```

```
program nome2;
{exemplo 2}
var a: real;
var b: integer;
procedure nomep(x: real);
var a, c: integer;
begin
read(c, a);
if a<x+c then
begin
a:= c+x;
write(a);
end
else c:= a+x;
end;
begin {programa principal}
read(b);
nomep(b);
end.
```