

Gramática

<PROGRAMA>	program <IDENTIFICADOR> ; <CORPO>
<IDENTIFICADOR>	Seqüência alfanumérica iniciada por char (tratado no lexico)
<CORPO>	<DECLARACOES> <BLOCO>
	<BLOCO>
<DECLARACOES>	<DEF_CONST> <DEF_TIPO> <DEF_VAR> <DEF_FUNC>
	€
<DEF_VAR>	var <VARIABLE> ; <VARIABLEIS>
	€
<DEF_CONST>	const <CONSTANTE> ; <CONSTANTES>
	€
<DEF_TIPOS>	type <TIPO> ; <TIPOS>
	€
<DEF_FUNC>	function <FUNCAO> <FUNCOES>
	€
<CONSTANTES>	<CONSTANTE> ; <CONSTANTES>
	€
<CONSTANTE>	<IDENTIFICADOR> = <CONST_VALOR>
<CONST_VALOR>	“ sequencia alfanumérica “
	<NOME_NUMERO> <EXP_MATEMATICA>
<NUMERO>	seqüência numérica
<TIPOS>	<TIPO> ; <TIPOS>
	€
<TIPO>	<IDENTIFICADOR> = <TIPO_DADO>
<TIPO_DADO>	integer
	real
	array [<NUMERO>] of <TIPO_DADO>
	<IDENTIFICADOR>
<VARIABLEIS>	<VARIABLE> ; <VARIABLEIS>
	€
<VARIABLE>	<IDENTIFICADOR> <LISTA_ID> : <TIPO_DADO>

<LISTA_ID>	, <IDENTIFICADOR> <LISTA_ID>
	€
<FUNCOES>	<FUNCAO> <FUNCOES>
	€
<FUNCAO>	<NOME_FUNCAO> <BLOCO_FUNCAO>
<NOME_FUNCAO>	<TIPO_DADO> <IDENTIFICADOR> (<VARIABLEIS>)
<BLOCO_FUNCAO>	<DEF_VAR> <BLOCO>
	<BLOCO>
<BLOCO>	begin <COMANDOS>
	<COMANDO> ;
<COMANDOS>	<COMANDO> ; <COMANDOS>
	end
<COMANDO>	<NOME> := <VALOR>
	while <EXPRESSAO_LOGICA> <BLOCO>
	if <EXPRESSAO_LOGICA> then <BLOCO> <SENAO>
	write <CONST_VALOR>
	read <NOME>
<SENAO>	else <BLOCO>
	€
<VALOR>	<IDENTIFICADOR> <VALOR_2>
	<NUMERO> <EXP_MATEMATICA>
<VALOR_2>	(<PARAMETRO>)
	<INDICE> <EXP_MATEMATICA>
<PARAMETRO>	<NOME_NUMERO> <LISTA_PARAM>
<LISTA_PARAM>	, <PARAMETRO>
	€
<EXP_LOGICA>	<NOME_NUMERO> <EXP_MATEMATICA> <EXP_LOGICA_2>
<EXP_LOGICA_2>	<OP_LOGICO> <EXP_LOGICA>
	€
<EXP_MATEMATICA>	<OP_MATEMATICO> <NOME_NUMERO> <EXP_MATEMATICA>
	€

<OP_LOGICO>	>		<		=		!
<OP_MATEMATICO>	+		-		*		/
<NOME_NUMERO>	<NOME>						
	<NUMERO>						
<NOME>	<IDENTIFICADOR>		<INDICE>				
<INDICE>	[<NOME_NUMERO>]						
	€						

```

program funcoes;
const
    TAM = 10;
type
    vetor = array[15] of integer;
var
    A, B, C, D : integer;
    E : vetor;

```

```

function
integer fatorial(a:integer;)
var i : integer;
begin
    i := 1;
    result:=1;
    while i < a
    begin
        result:=result*i;
        i:=i+1;
    end;
end

```

```

real exp(a: real; b: real;)
var i : integer;
begin
    i := 1;
    result := a;
    if b = 0 then
        result := 1;
    else
        while i < b
        begin
            result := a * a;
            i := i + 1;
        end
    end

```

```

integer maior(a : vetor;)
var i : integer;
begin
    i := 0;
    result := a[0];
    while i < 15
    begin
        if a[i] > result then
            result := a[i];
        end;
    end

    begin
        A:=TAM;
        B := fatorial(A);
        C := exp(A,B);
        D := maior(E);
    end

```