

**Nome:** João Marcos Salvanini Bellini de Moraes

**Nome da Linguagem:** PP (Programe em Português)

**Linguagem desenvolvimento:** Java

**Introdução:**

A linguagem PP tem como objetivo auxiliar aqueles com dificuldade em aprender linguagens de programação por não terem conhecimento do inglês. O PP é uma linguagem procedural semelhante à linguagem C, mas executa somente as instruções mais básicas, tais como operações matemáticas, estruturas condicionais e laços de repetição.

**Tabela de Tokens:**

Token	Lexema	Regex	Descrição
var	x, xy, x123y, x_y, XY5, _x12, _1, x1_, ...	[a-zA-Z_] [a-zA-Z0-9_]*	Qualquer variável dentro do código
comentario	# comentario #	\# [^\#]* \#	Comentário dentro do código
numero	0..9	\d+	Valores numéricos no código
se	se	se	Palavra reservada “se” - se condicional
sese	sese	sese	Palavra reservada “sese” - condicional senão se
senao	senao	senao	Palavra reservada “senao” - condicional
tipo	inteiro caractere logico texto	inteiro caractere logico texto	Palavras reservadas “inteiro”, “caractere”, “logico”, “texto” - tipos de dados
booleano	verdadeiro falso	verdadeiro falso	Palavras reservadas “verdadeiro” e “falso” - tipo de dado booleano
enquanto	enquanto	enquanto	Palavra reservada “enquanto” - para laços de repetição
ap	(	\ (	Abre parênteses

fp	)	\)	Fecha parênteses
ach	{	\{	Abre chaves
fch	}	\}	Fecha chaves
quebra_linha	;	;	Final de linha
operador	*, /, +, -	[ \* \ / \ + \ - ]   resto   e   ou	Operadores
atribuicao	=	=	Operador de atribuição
virgula	,	\,	Vírgula
comparador	==, !=, >, <, >=, <=	[ < > ] = ?   [ ! = ] =	Comparadores
leia	leia	leia	Função implementada “leia” - para ler variáveis
mostre	mostre	mostre	Função implementada “mostre” - para escrever na tela
caractere	‘a’, ‘2’, ‘X’	\ ' [ a - z A - Z 0 - 9 ] \ '	Caractere dentro do código
texto	“ação”	\ " [ ^ \ " ] * \ "	Texto dentro do código

### Gramática:

```

programa      ::= ACH corpo FCH
                | ACH FCH;

corpo         ::= corpo conteudo
                | conteudo;

conteudo      ::= exp          QUEBRA_LINHA
                | atrib        QUEBRA_LINHA
                | decl         QUEBRA_LINHA
                | leitura      QUEBRA_LINHA
                | saida        QUEBRA_LINHA
                | se
                | se_sese
                | se_senao

```

	se_sese_senao
	repet
	COMENTARIO
	programa;
val	::= NUMERO
	BOOLEANO
	CARACTERE
	VAR;
exp	::= exp OPERADOR exp
	exp COMPARADOR exp
	AP exp FP
	val;
decl	::= TIPO VAR
	TIPO atrib;
atrib	::= VAR ATRIBUICAO exp;
leitura	::= LEIA AP params FP;
params	::= VAR
	VAR mais_params;
mais_params	::= VIRGULA VAR
	VIRGULA VAR mais_params;
saida	::= MOSTRE AP TEXTO VIRGULA params FP;
se	::= SE AP exp FP programa;
se_sese	::= se n_sese;
n_sese	::= SESE AP exp FP programa
	SESE AP exp FP programa n_sese;
se_senao	::= se SENAO programa;
se_sese_senao	::= se_sese SENAO programa;
repet	::= ENQUANTO AP exp FP programa;