



Transcrição de aula

Disciplina	Processamento de Linguagens - 3º ano - LEI	
Secretário	Número: 7433	Nome: Delfim Macedo
Data:	2011-04-28	Nº Página 3
Turno:	TP3	Nº Alunos 17

SUMÁRIO

Introdução de ações semânticas em parsers codificados em yacc.

[]

[]

[, 22, 7]

Construir a Gramática com recursividade à esquerda.

Lista \rightarrow "[seq "]" | "[]"

Seq \rightarrow num | Seq ',' num

% {

#include "y.tab.h"

% }

%-Y-

%-Y-

```
{ yyval.rint = atoi (yytext); return NUM; }
```



Transcrição (folha de continuação)

```
%a e %b { int cont=0; soma=0 %b
% toku NUM ERRO
% %p <int> NUM
% union
{
  int vint;
}
%
%
z = Lista l; { printf("Somos %d; Elementos %d\n",
               soma, cont); }
Lista = [' ' ' ' ] { printf("Lista vazia.\n"); }
        [' seq ' ] " " ...
        ;
seq = NUM { cout << " " ; soma = soma + 1; printf("seq = %d\n", seq); }
        [' seq ' , ' NUM " " ...
        ;
%
int %error (char *s) {
  printf("ERRO: %s; toku: %s\n", s, %text);
}
%
```

Transcrição (folha de continuação)

É um máximo {
yy para $SO(L)$;

g

—k—

Se gerar uma gramática com Expressões
Aritméticas, com recursividade a
esquerda

$$2 + 3$$

$$4 + 2 * 3$$

$$5 + (2 - 3) * 2$$

$$\text{lista} \rightarrow []$$

$$| \text{[seq]}$$

$$\text{seq} \rightarrow \text{num}$$

$$| \text{seq}, \text{num}$$