

# Documentação do Trabalho - Compiladores (ECOM06A)

Nome: Bruno Corrêa Silva - 2023002968  
Filipi de Souza Furtado - 2023005315  
Luiz Felipe Carneiro Pontes - 2023005638

## 1. Expressões Regulares

Nesta seção estão definidas as expressões regulares para o reconhecimento de cada token da linguagem criada.

Elemento	Lexema em C	Lexema Novo	Expressão Regular	Token
Identificadores		"var_1"	[a-zA-Z_][a-zA-Z0-9_]*	ID
Palavras-chave	int	o_anuncio_dizia_tres_mil		T_INT
	float	aluguel?		T_FLOAT
	char	e_so_um_pico		T_CHAR
	scanf	pode_falar		T_SCAN
	printf	<a href="#">deixa a mamee e o papai conversarem</a>		T_PRINT
	if	<a href="#">com grandes poderes</a>		T_IF
	else	vem_grandes_responsabilidades		T_ELSE
	else if	<a href="#">se alguem comesse promessa minha filha seria gorda</a>		T_ELSEIF
	switch	<a href="#">esta um lixo</a>		T_SWITCH
	case	lixo		T_CASE
	default	super_lixo		T_DEFAULT
	break	<a href="#">me quebrei</a>		T_BREAK
	continue	voltei		T_CONTINUE
	while	<a href="#">aguenta ai mary jane</a>		T_WHILE
	for	dia_apos_dia		T_FOR

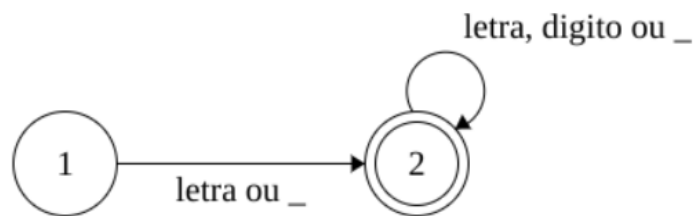
Inteiros	"-200"		[+]?[0-9]+	INT_CONST
Floats	"3.14"		[+]?[0-9]+.[0-9]+	FLOAT_CONST
Chars	"a"		'[a-zA-Z]'	CHAR_CONST
Strings	"frase A"		"[^\"\\n]*"	STRING_CONST
Operadores aritméticos	+	posso_fazer_mais		PLUS_OP
	-	eu_perdi_o_tio_ben		MINUS_OP
	*	potencializa_as_caracteristicas		MULT_OP
	/	ele_roubou_o_cara_das_pizzas		DIV_OP
	mod	quem_disse_que_isso_e_problema_me u		MOD_OP
Operadores relacionais	==	voce_e_como_eu		EQ_OP
	!=	voce_nao_e_como_eu		NE_OP
	>=	voce_e_mais_forte_do_que_pensa		GE_OP
	<=	ainda_nao_sou_tudo_que_posso_ser		LE_OP
	>	voce_esta_acima_disso		GT_OP
	<	voce_ainda_tem_muito_o_que_aprende r		LT_OP
Operadores Lógicos	&&	juntos_podemos_conquistar_tudo		AND_OP
		tem_sempre_outra_maneira		OR_OP
	!	isso_nao_e_verdade		NOT_OP
Atribuição	=	eu_sou_outra_pessoa		ASSIGN_OP
Ponto e vírgula	;			SEMICOLON
Vírgula	,			COMMA
Parênteses	(			LPAREN
	)			RPAREN
Chaves	{			LBRACE
	}			RBRACE
Comentários (linha)	//Comentário		/*.*	LINE_COMMENT

Comentários (bloco)	/*  */	Comentário	/\*[\s\S]+\*/	BLOCK_COMMENT
------------------------	--------------	------------	---------------	---------------

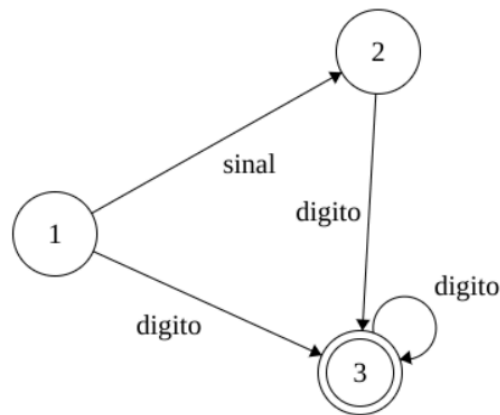
## 2. Autômatos Finitos Das Expressões

Aqui estão diagramas representando os Autômatos Finitos Determinísticos (AFDs) construídos a partir das expressões regulares. O estado inicial de todos eles será sempre o **1**.

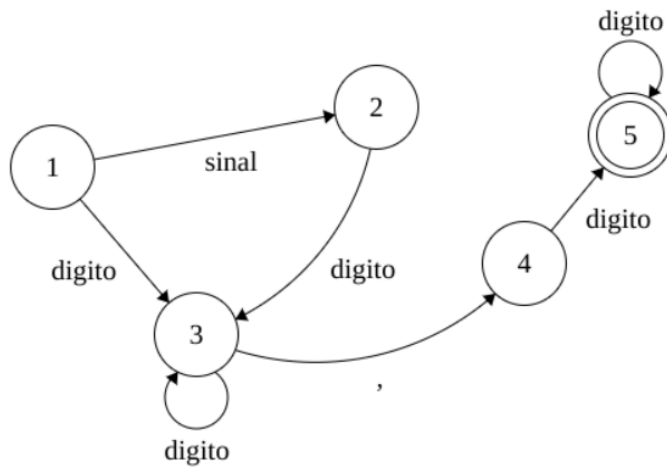
### Identificadores



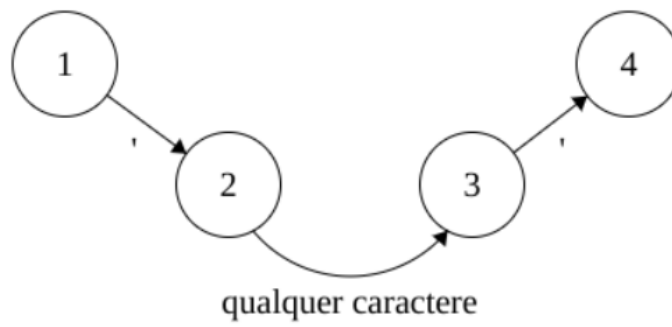
### Inteiros



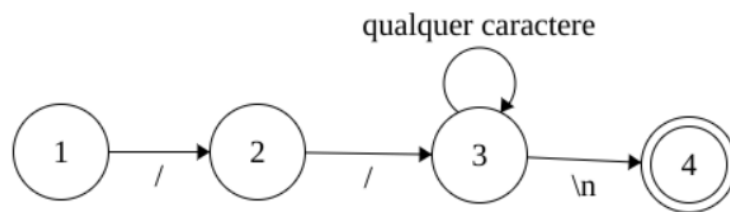
### Floats



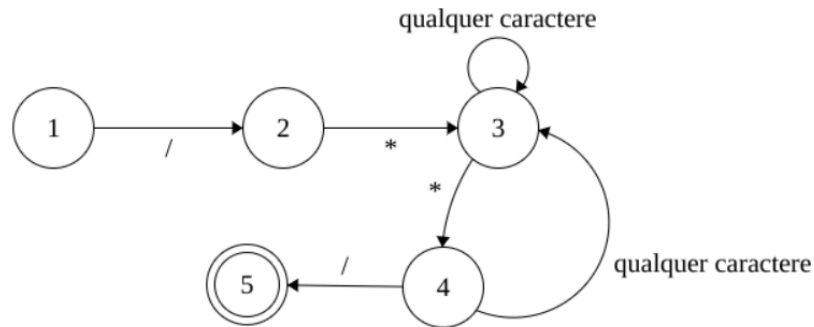
### Chars



### Comentário (linha)



### Comentário (bloco)



### 3. Gramática da Linguagem

Abaixo estão as produções utilizadas para o reconhecimento dos comandos da linguagem.

Símbolos Não-Terminais:

- **S**: Início do programa
- **DECL**: Declaração ou inicialização
- **CMD**: Comando
- **EXPR**: Expressão
- **TIPO**: Tipo de dado
- **SE, COND, ENQUANTO, PARA, QUEBRA, CONTINUA**: Estruturas de controle
- **LE, ESCREVE**: Entrada e saída

Produções:

$S \rightarrow DECL\ S \mid CMD\ S \mid \varepsilon$

$DECL \rightarrow TIPO\ ID\ ';' \mid TIPO\ ID\ '='\ EXPR\ ';'$

$TIPO \rightarrow 'o\_anuncio\_dizia\_tres\_mil' \mid 'aluguel?' \mid 'e\_so\_um\_pico'$

$CMD \rightarrow SE \mid COND \mid ENQUANTO \mid PARA \mid EXPR\ ';' \mid QUEBRA\ ';' \mid CONTINUA\ ';' \mid LE \mid ESCREVE \mid \{'\ S\ '\}$

SE → 'com\_grandes\_poderes' '(' EXPR ')' CMD  
( 'se\_alguem\_comesse\_promesa,minha\_filha\_seria\_gorda' '(' EXPR ')' CMD)\*  
( 'vem\_grandes\_responsabilidades' CMD)?

COND → 'esta\_um\_lixo' '(' EXPR ')' '{' CASOS '}'

ENQUANTO → 'aguenta\_ai\_mary\_jane' '(' EXPR ')' CMD

PARA → 'dia\_apos\_dia' '(' EXPR | DECL ';' EXPR ';' EXPR ')' CMD

QUEBRA → 'me\_quebrei'

CONTINUA → 'voltei'

EXPR → ID '=' EXPR

| EXPR OP EXPR

| '(' EXPR ')'

| NUM

| ID

OP → ARITH\_OP | RELAT\_OP

RELAT\_OP → EQ\_OP | NE\_OP | LT\_OP | LE\_OP | GT\_OP | GE\_OP

NUM → INT\_CONST | FLOAT\_CONST | CHAR\_CONST

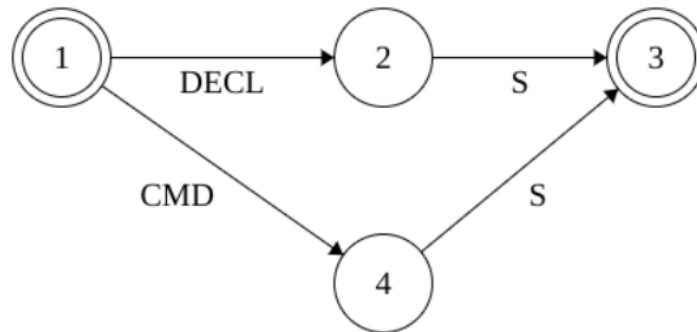
CASOS → 'lixo' NUM ':' S CASOS | 'super\_lixo' ':' S CASOS |  $\epsilon$

LE → 'pode\_falar' '(' STRING\_CONST (',' EXPR)\* ')' ';' ;

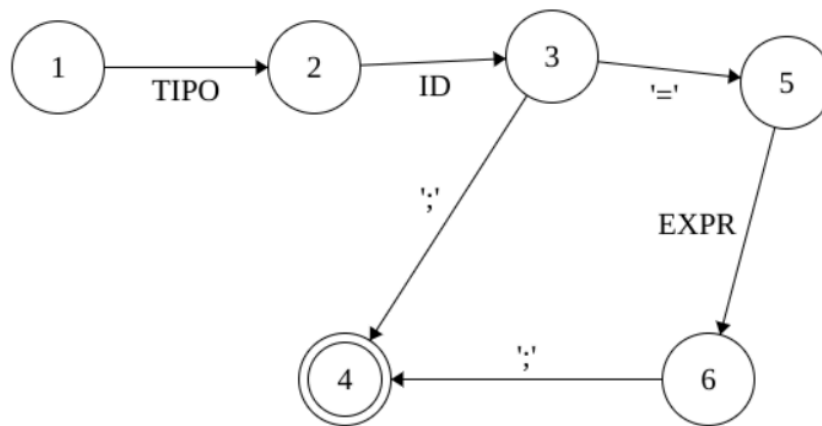
ESCREVE → 'deixa\_a\_mamae\_e\_o\_papai\_conversarem' '(' STRING\_CONST (',' EXPR)\* ')' ';' ;

#### 4. Autômatos das Regras de Produção

S



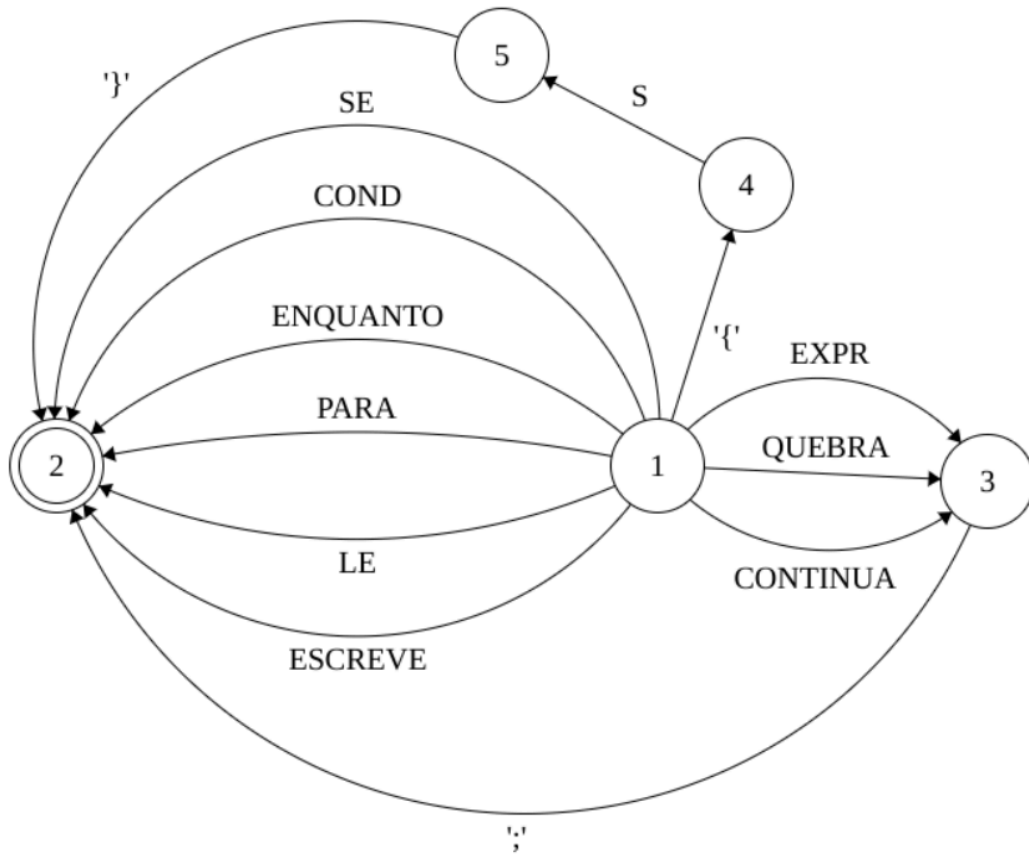
DECL



TIPO

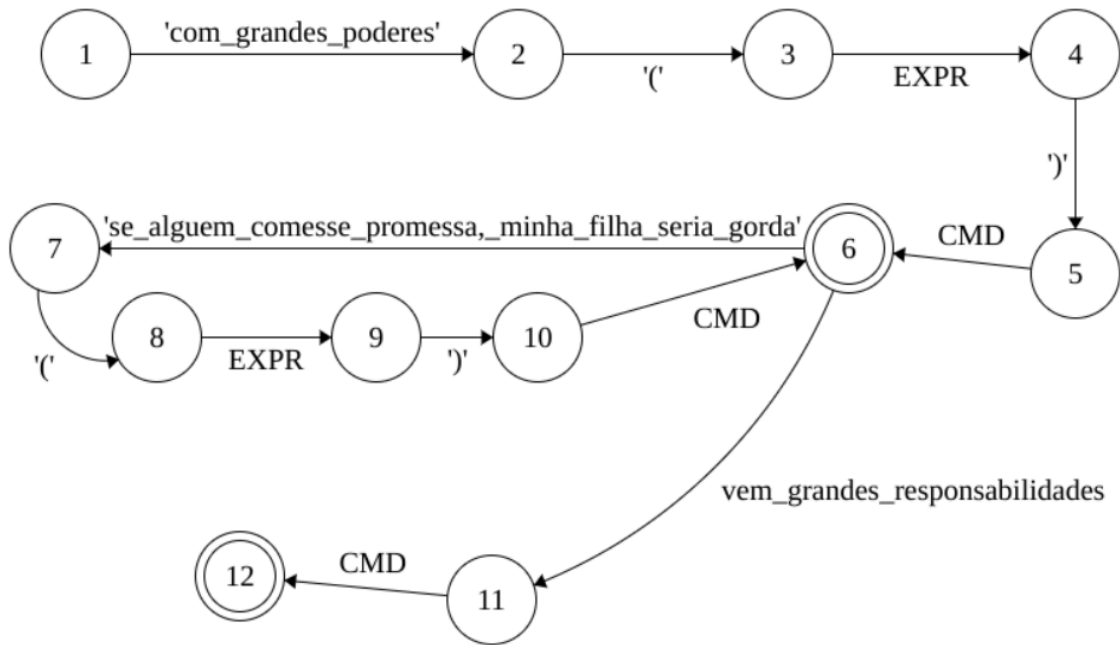


CMD

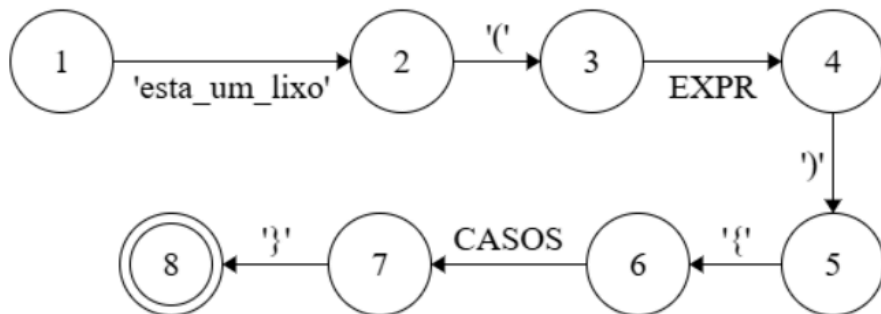




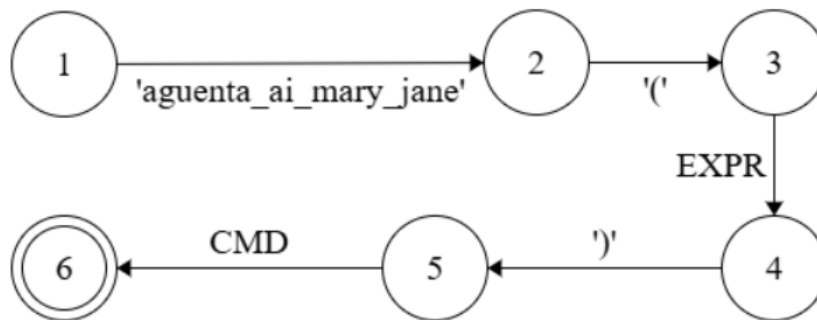
## SE



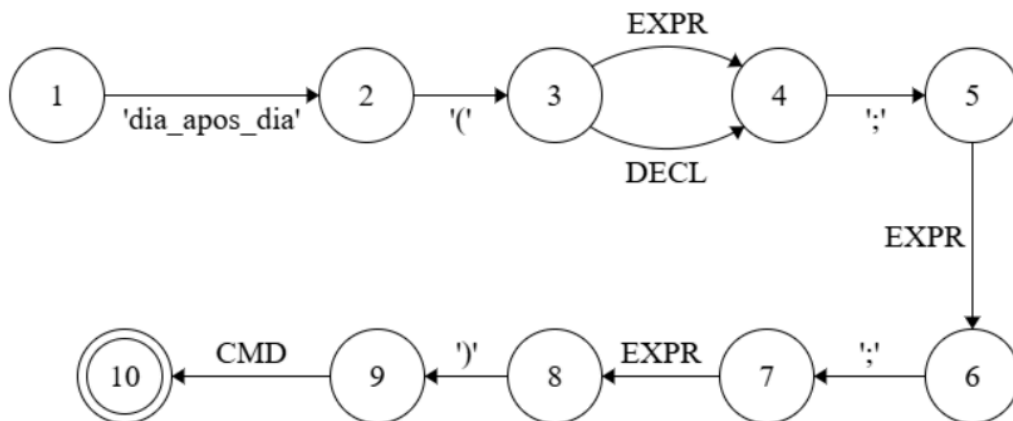
## COND



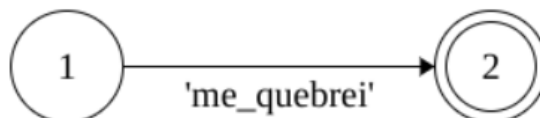
### ENQUANTO



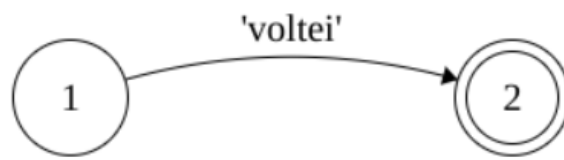
### PARA



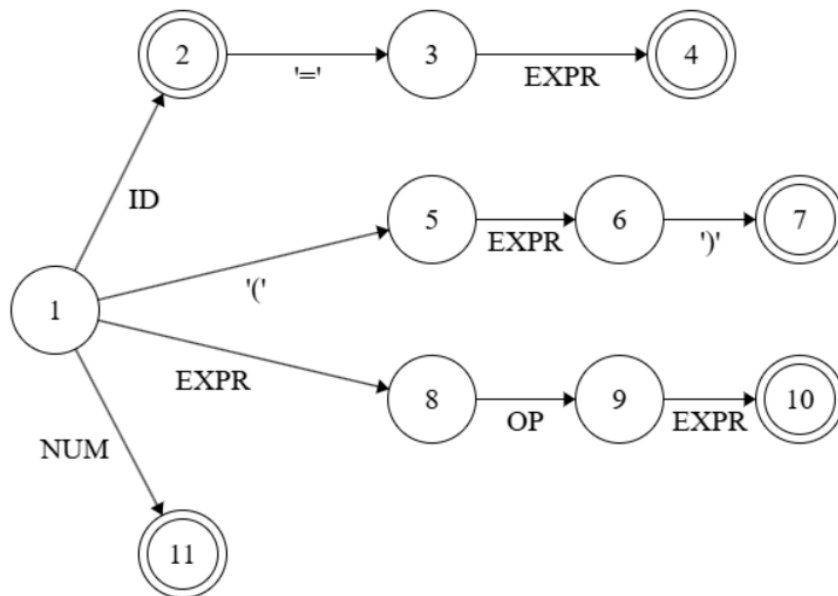
### QUEBRA



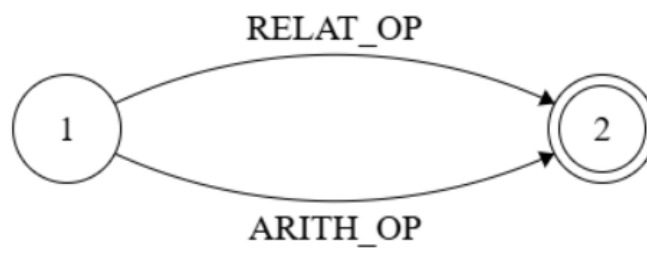
CONTINUA



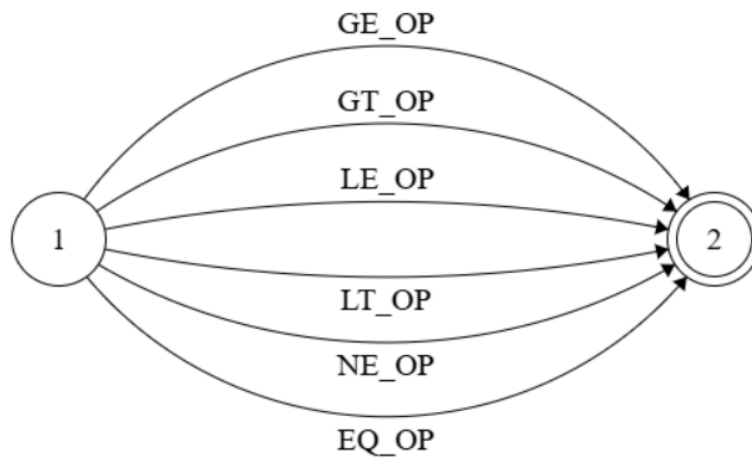
EXPR



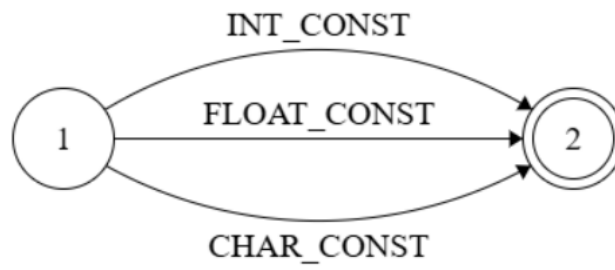
OP



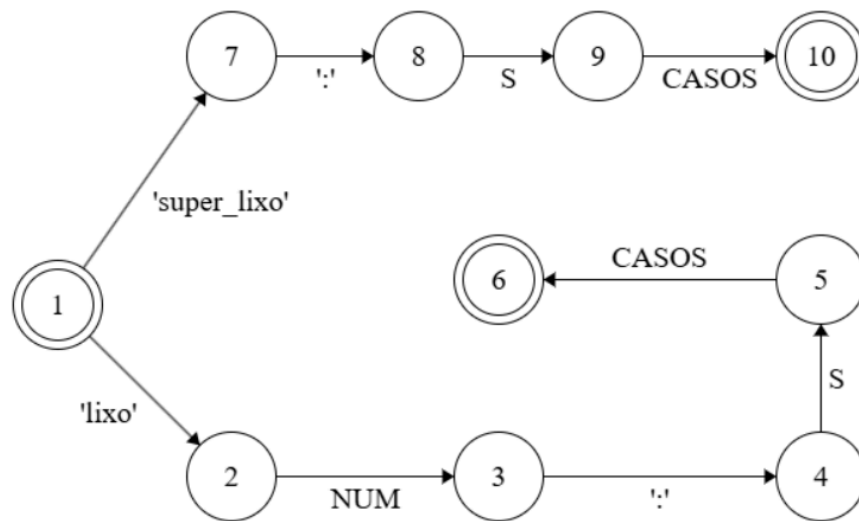
### RELAT\_OP



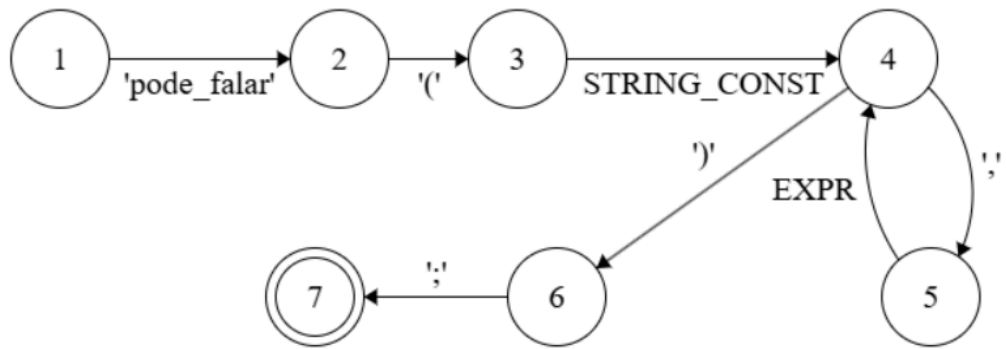
### NUM



### CASOS



LE



ESCREVE

