

André Marcos Leifeld Raicoski,
Ryan Felipe Ferreira Ribeiro,
Thiago Batista Araújo

TypUai

Brasil

Outubro de 2023

André Marcos Leifeld Raicoski,
Ryan Felipe Ferreira Ribeiro,
Thiago Batista Araújo

TypUai

Documentação referente ao Trabalho avaliativo da disciplina ECOM06A - Compiladores do curso de Engenharia da Computação da Universidade Federal de Itajubá

Universidade Federal de Itajubá – UNIFEI
Instituto de Engenharia de Sistemas e Tecnologia da Informação
Engenharia da Computação

Orientador: Thatyana Seraphim

Brasil
Outubro de 2023

Lista de ilustrações

Figura 1 – Autômato Tipo de Dado - intero	11
Figura 2 – Autômato Tipo de Dado - real	11
Figura 3 – Autômato Tipo de Dado - caractere	11
Figura 4 – Autômato Tipo de Dado - bool	12
Figura 5 – Autômato Tipo de Dado - Nome variável	12
Figura 6 – Autômato Operações Relacionais - igualdade ou atribuição	12
Figura 7 – Autômato Operações Relacionais - maior ou maior ou igual	12
Figura 8 – Autômato Operações Relacionais - menor ou menor ou igual	13
Figura 9 – Autômato Operações Relacionais - diferente	13
Figura 10 – Autômato Operador Aritmético - soma	13
Figura 11 – Autômato Operadores Aritmético - subtração	13
Figura 12 – Autômato Operador Aritmético - multiplicação	13
Figura 13 – Autômato Operador Aritmético - divisão	14
Figura 14 – Autômato Operador Aritmético - resto	14
Figura 15 – Autômato Operador Aritmético - inverte	14
Figura 16 – Autômato Operações Relacionais - operador e	14
Figura 17 – Autômato Operações Relacionais - operador ou	14
Figura 18 – Autômato Operações Relacionais - operador negação	15
Figura 19 – Autômato Palavras Reservadas - intero	15
Figura 20 – Autômato Palavras Reservadas - real	15
Figura 21 – Autômato Palavras Reservadas - caractere	15
Figura 22 – Autômato Palavras Reservadas - bool	15
Figura 23 – Autômato Palavras Reservadas - caso	16
Figura 24 – Autômato Palavras Reservadas - naodeu	16
Figura 25 – Autômato Palavras Reservadas - ese	16
Figura 26 – Autômato Palavras Reservadas - floop	16
Figura 27 – Autômato Palavras Reservadas - uai	16
Figura 28 – Autômato Palavras Reservadas - retorna	16
Figura 29 – Autômato Abertura de Parênteses	17
Figura 30 – Autômato Caracter Arroba	17
Figura 31 – Autômato Caracter Aspas Duplas	17
Figura 32 – Autômato Caracter Acento Circunflexo	17
Figura 33 – Autômato Caracter Dois Pontos	17
Figura 34 – Autômato Fechar Parênteses	17
Figura 35 – Autômato Caracter Ponto	18
Figura 36 – Autômato Caracter Ponto e Vírgula	18

Figura 37 – Autômato Character Vírgula	18
Figura 38 – Autômato Fechar Bloco de Comando ou Interromper	18
Figura 39 – Autômato Início de Bloco de Comando	18
Figura 40 – Autômato - OP	19
Figura 41 – Autômato - ID	19
Figura 42 – Autômato - EXP_REAL	20
Figura 43 – Autômato - EXP_INT	21
Figura 44 – Autômato - EXP_CHAR	22
Figura 45 – Autômato relacionado à saída	22
Figura 46 – Autômato relacionado à entrada	22
Figura 47 – Autômato - Operadores Relacionais	23
Figura 48 – Autômato - Expressões Relacionais	23
Figura 49 – Autômato - Operadores Logicos	24
Figura 50 – Autômato - Expressões Lógicas	24
Figura 51 – Autômato - retorna	25
Figura 52 – Autômato das Estruturas Condicionais	25
Figura 53 – Autômato de Criação de Funções	26
Figura 54 – Autômato de Chamada de Funções	27
Figura 55 – Autômato de Estruturas de Repetição	27

Lista de tabelas

Tabela 1 – Tipos de dados - TypUai	7
Tabela 2 – Comandos - TypUai	7
Tabela 3 – Operadores Relacionais - TypUai	8
Tabela 4 – Operadores Aritméticos - TypUai	8
Tabela 5 – Operadores Lógicos - TypUai	8
Tabela 6 – Simbolos Especiais TypUai	9
Tabela 7 – Blocos de Comando - TypUai	9
Tabela 8 – Palavras Reservadas - TypUai	9
Tabela 9 – Relação Comando x Token - TypUai	10

Sumário

1	INTRODUÇÃO	6
2	ANALISADOR LÉXICO	7
2.1	Tipos de dados	7
2.2	Comandos	7
2.3	Operadores Relacionais	8
2.4	Operadores Aritméticos	8
2.5	Operadores lógicos	8
2.6	Simbolos especiais	9
2.7	Blocos de comandos	9
2.8	Palavras reservadas	9
2.9	Tokens	10
2.10	Autômatos	11
2.10.1	Tipos de dados	11
2.10.2	Operadores Relacionais	12
2.10.3	Operadores Aritméticos	13
2.10.4	Operadores Lógicos	14
2.10.5	Palavras reservadas	15
2.10.6	Símbolos especiais	17
2.10.7	Blocos de comando	18
3	ANALISADOR SINTÁTICO	19
3.1	Expressões Aritméticas	19
3.2	Entradas e saídas	22
3.3	Expressões Relacionais	23
3.4	Expressões Logicas	24
3.5	Declarando Variáveis	25
3.6	Estruturas Condicionais	25
3.7	Funções	26
3.8	Estruturas de Repetição	27

1 Introdução

A análise léxica e sintática desempenham um papel crucial na interpretação e execução de linguagens de programação ou na compreensão de estruturas linguísticas em contextos naturais. Este projeto oferece a oportunidade de explorar e aplicar os princípios teóricos na construção de analisadores, para que os conhecimentos passados na disciplina ECOM06A - Compiladores sejam trabalhados. Para isso, os analisadores arquitetados foram projetados baseados em vontades e características que os integrantes dos grupos mais se identificam. Assim, nasce a prototipação de uma nova linguagem: o TypUai.

2 Analisador Léxico

2.1 Tipos de dados

Inteiro	intero	$([-]?[0-9]+)$
Real	real	$([0-9]+([0-9]+)?)$
Booleano	bool	$([8] [80])$
Caractere	caractere	$('[a-zA-Z0-9]')$
Variável	nomeVar	$([a-z]([a-zA-Z0-9]*)?)$

Tabela 1 – Tipos de dados - TypUai

De maneira simplificada, com base nas expressões regulares a seguir, pode-se definir as características de um dígito, letra e nome de variável, respectivamente.

* $([0-9])$

* $([A-Za-z])$

* $([letra] + [dígito|letra])$

Sendo que o nome de variável, nomeVar, obedece a regra estabelecida na tabela 1.

2.2 Comandos

Entrada	<-
Saída	->
Atribuição	=
Se	caso
Senão	naodeu
Senão se	ese
Para	floop
Enquanto	uai

Tabela 2 – Comandos - TypUai

2.3 Operadores Relacionais

Equivalente	==
Maior ou igual	>=
Menor ou igual	<=
Maior	>
Menor	<
Diferente	!!

Tabela 3 – Operadores Relacionais - TypUai

2.4 Operadores Aritméticos

Adição	+
Subtração	-
Multiplicação	*
Divisão	/
Troca sinal	~
Resto	%

Tabela 4 – Operadores Aritméticos - TypUai

2.5 Operadores lógicos

E	"e"
Ou	"ou"
Negação	"!"

Tabela 5 – Operadores Lógicos - TypUai

2.6 Simbolos especiais

Ponto	.
Dois pontos	:
Ponto e vírgula	;
Abre parênteses	(
Fecha parênteses)
Aspas Duplas	"
Arroba	@
Vírgula	,
Circunflexo	^

Tabela 6 – Simbolos Especiais TypUai

2.7 Blocos de comandos

Início	:
Fim	;
Break	"vaza "

Tabela 7 – Blocos de Comando - TypUai

2.8 Palavras reservadas

intero
real
caractere
bool
caso
naodeu
ese
floop
uai
no
retorna

Tabela 8 – Palavras Reservadas - TypUai

2.9 Tokens

Comando	Token
:	comeco
;	fim
intero	int
real	float
caractere	char
bool	booleano
<-	entrada
->	saida
=	atribuicao
caso	se
naodeu	senao
ese	senaose
floop	para
uai	enquanto
==	igual
>=	maiorigual
>	maior
<=	menorigual
<	menor
!!	diferente
+	soma
-	subtracao
/	divisao
~	inverte
%	resto
*	multiplicacao
e	e
ou	ou
!	nao
.	ponto
;	pontovirgula
(abreparenteses
)	fechaparenteses
"	aspadupla
@	arroba
^	fimcomando
no	no
,	virgula
retorna	retorna

Tabela 9 – Relação Comando x Token - TypUai

2.10 Autômatos

2.10.1 Tipos de dados

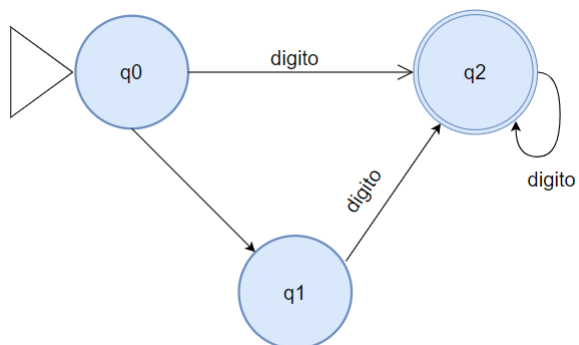


Figura 1 – Autômato Tipo de Dado - inteiro

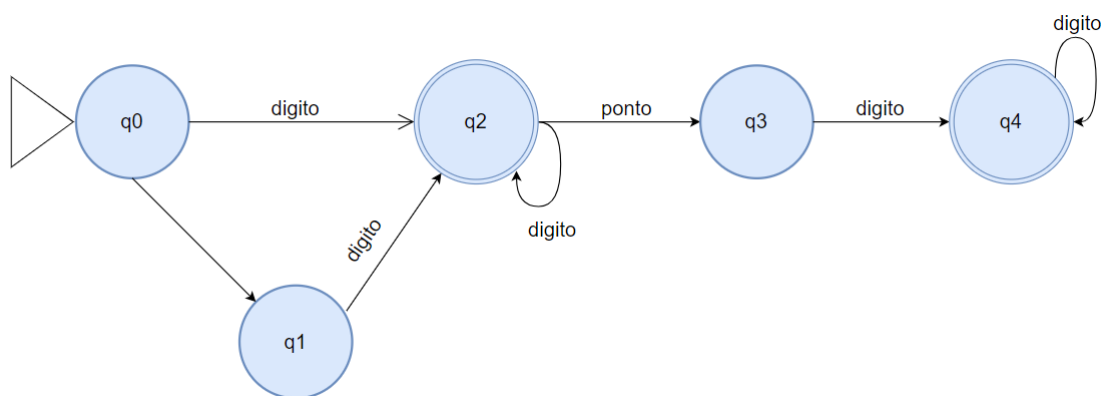


Figura 2 – Autômato Tipo de Dado - real

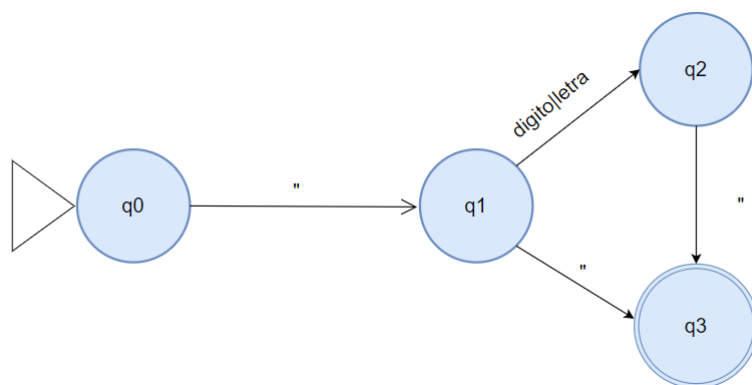


Figura 3 – Autômato Tipo de Dado - caractere

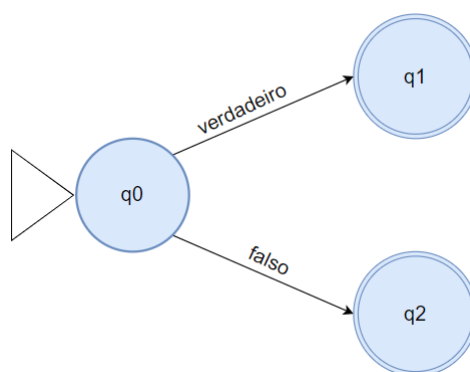


Figura 4 – Autômato Tipo de Dado - bool

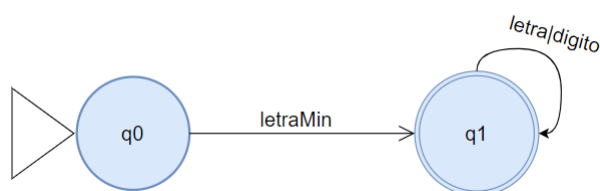


Figura 5 – Autômato Tipo de Dado - Nome variável

2.10.2 Operadores Relacionais

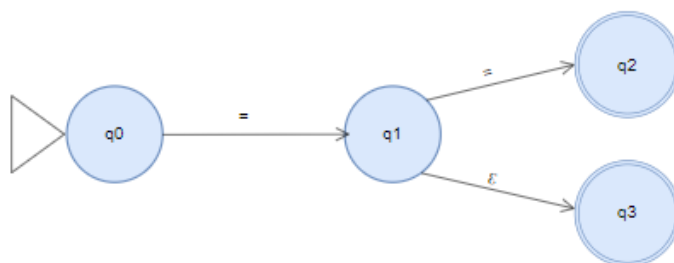


Figura 6 – Autômato Operações Relacionais - igualdade ou atribuição

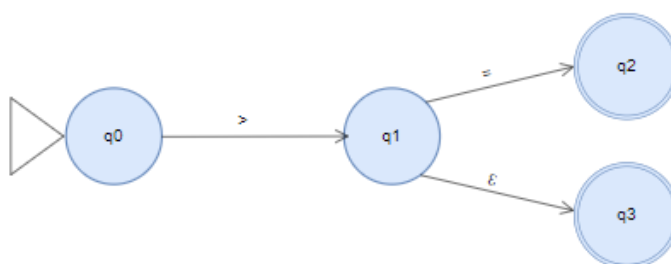


Figura 7 – Autômato Operações Relacionais - maior ou maior ou igual

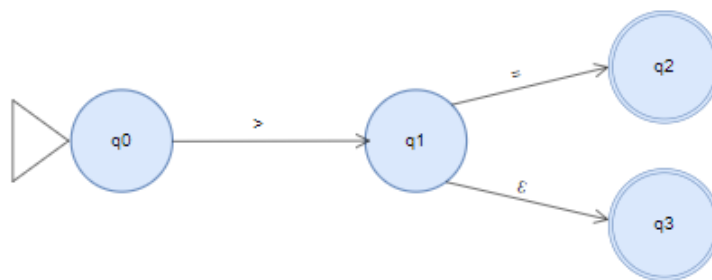


Figura 8 – Autômato Operações Relacionais - menor ou menor ou igual

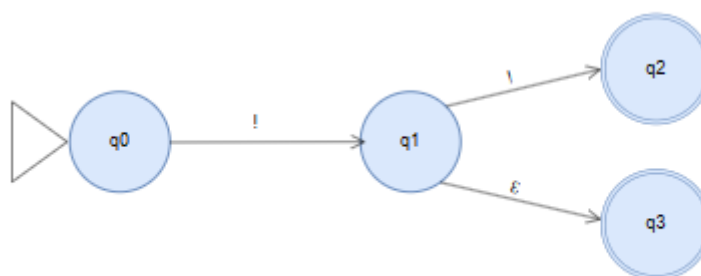


Figura 9 – Autômato Operações Relacionais - diferente

2.10.3 Operadores Aritméticos

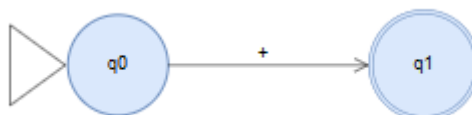


Figura 10 – Autômato Operador Aritmético - soma

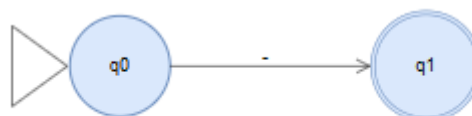


Figura 11 – Autômato Operadores Aritmético - subtração

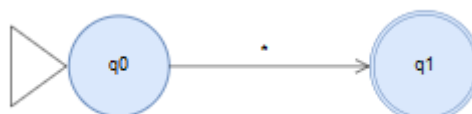


Figura 12 – Autômato Operador Aritmético - multiplicação

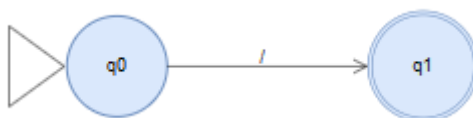


Figura 13 – Autômato Operador Aritmético - divisão

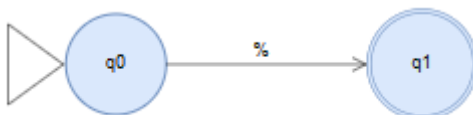


Figura 14 – Autômato Operador Aritmético - resto



Figura 15 – Autômato Operador Aritmético - inverte

2.10.4 Operadores Lógicos

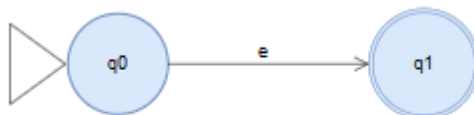


Figura 16 – Autômato Operações Relacionais - operador e

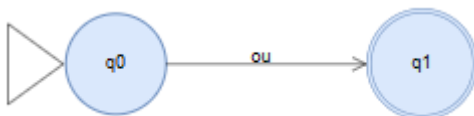


Figura 17 – Autômato Operações Relacionais - operador ou

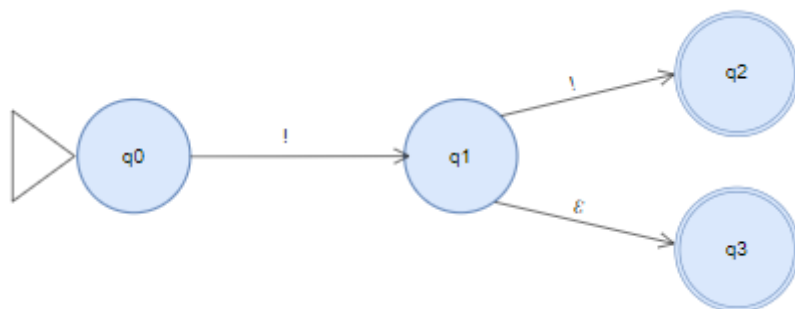


Figura 18 – Autômato Operações Relacionais - operador negação

2.10.5 Palavras reservadas

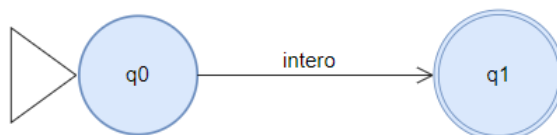


Figura 19 – Autômato Palavras Reservadas - intero

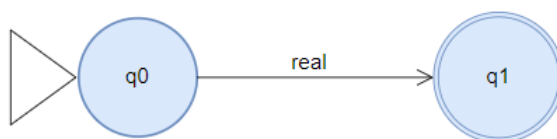


Figura 20 – Autômato Palavras Reservadas - real

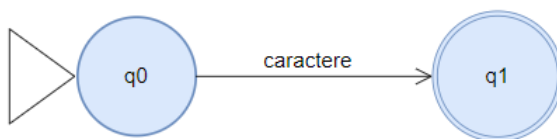


Figura 21 – Autômato Palavras Reservadas - caractere

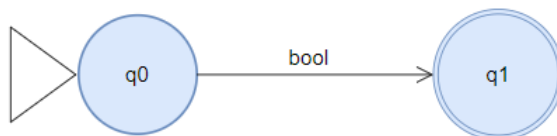


Figura 22 – Autômato Palavras Reservadas - bool

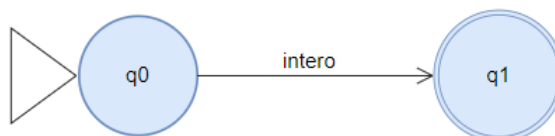


Figura 23 – Autômato Palavras Reservadas - caso

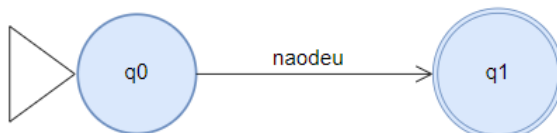


Figura 24 – Autômato Palavras Reservadas - naodeu

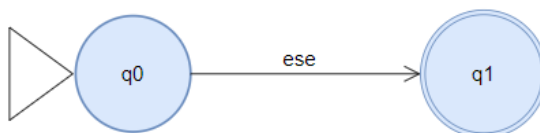


Figura 25 – Autômato Palavras Reservadas - ese

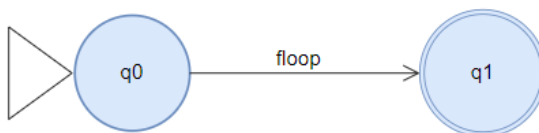


Figura 26 – Autômato Palavras Reservadas - floop

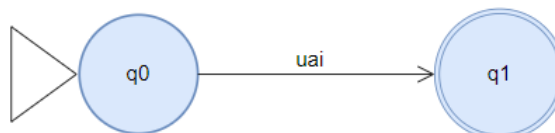


Figura 27 – Autômato Palavras Reservadas - uai

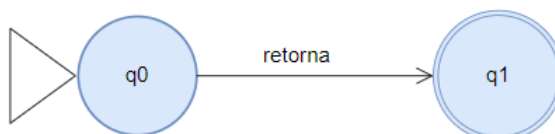


Figura 28 – Autômato Palavras Reservadas - retorna

2.10.6 Símbolos especiais

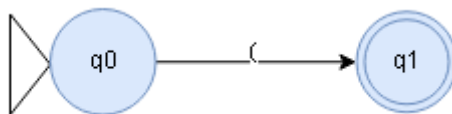


Figura 29 – Autômato Abertura de Parênteses

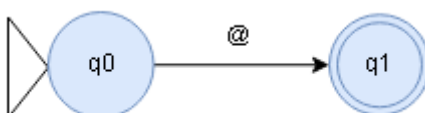


Figura 30 – Autômato Caracter Arroba

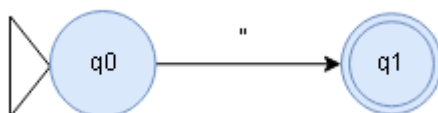


Figura 31 – Autômato Caracter Aspas Duplas



Figura 32 – Autômato Caracter Acento Circunflexo



Figura 33 – Autômato Caracter Dois Pontos

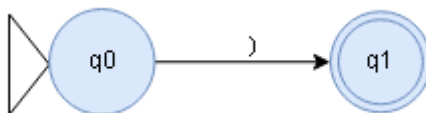


Figura 34 – Autômato Fechar Parênteses



Figura 35 – Autômato Caracter Ponto

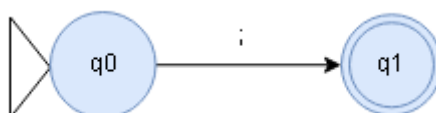


Figura 36 – Autômato Caracter Ponto e Vírgula

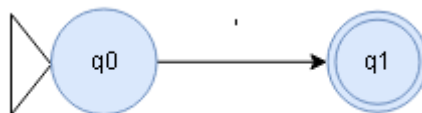


Figura 37 – Autômato Caracter Vírgula

2.10.7 Blocos de comando

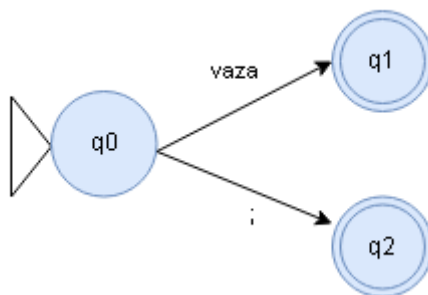


Figura 38 – Autômato Fechar Bloco de Comando ou Interromper

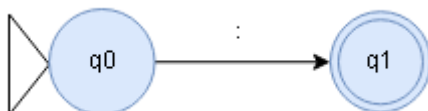


Figura 39 – Autômato Início de Bloco de Comando

3 Analisador Sintático

3.1 Expressões Aritméticas

Aqui, as regras para expressões aritméticas são determinadas. Note, abaixo, as definições de TypUai.

Para fins de facilitação, considere "operações aritméticas"OP.

OP := soma
 | subtracao
 | inverte
 | divisao
 | multiplicacao
 | resto

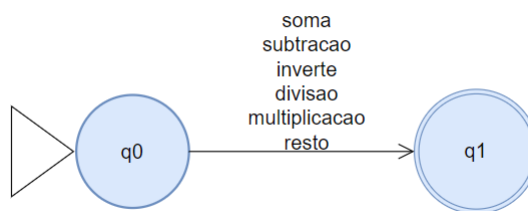


Figura 40 – Autômato - OP

ID := booleano
 | int
 | char
 | float

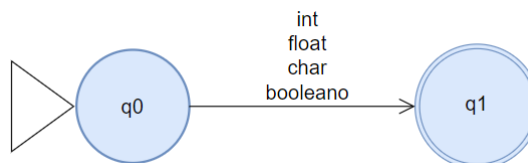


Figura 41 – Autômato - ID

A próxima norma aborda a interpretação de uma operação real. Nesta, as regras de expressão padrão envolvem a substituição por (expressão) e -expressão. Além disso, é

possível realizar operações entre números decimais e inteiros, com um deles sendo sempre decimal. Também possível substituir a expressão por um número decimal, pois o resultado é invariavelmente um número decimal.

EXP REAL := abreparenteses EXP_REAL fechaparenteses

| menos EXP_REAL

| float

| float:a OP float:b

| int:a OP float:b

| float:a OP int:b

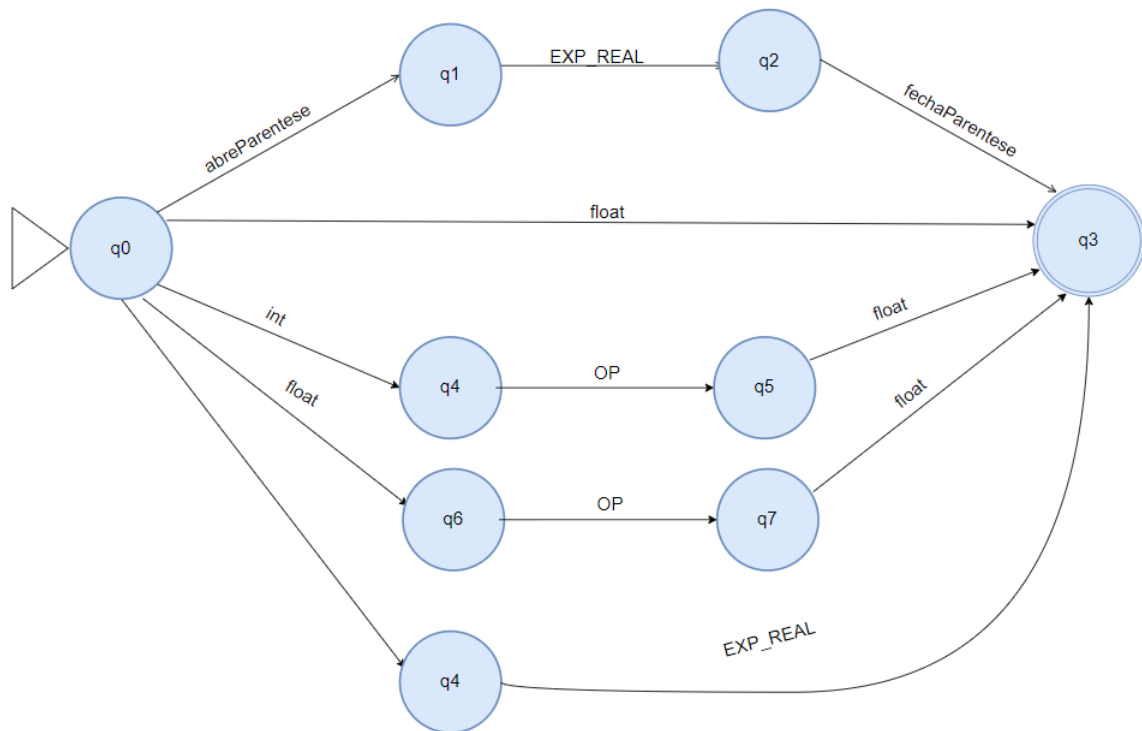


Figura 42 – Autômato - EXP_REAL

A regra subsequente diz respeito a uma operação inteira. Assim como a anterior, ela segue o padrão de regras de expressão. No entanto, ao contrário da operação real, aceita apenas números inteiros. É também possível substituir a expressão por um número inteiro.

EXP INT := abreparenteses EXP INT fechaparenteses | int
 | menos EXP INT
 | int:a OP int:b

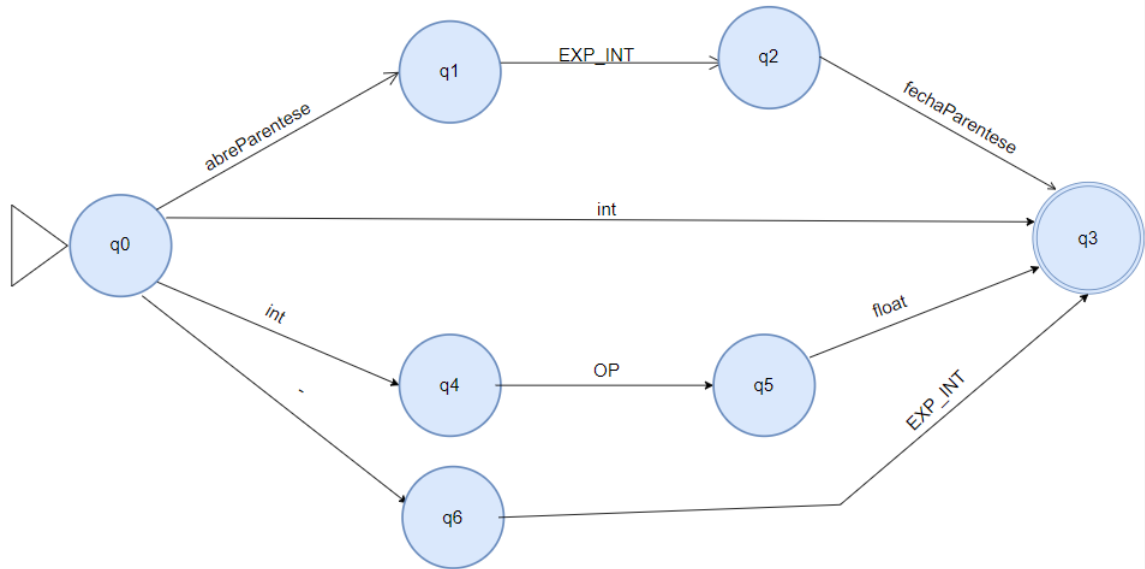


Figura 43 – Autômato - EXP_INT

A norma para expressões char apresenta uma pequena variação. Ela permite apenas operações de adição e subtração o tipo char. Além disso, aceita a substituição por (expressão), por um char.

EXP CHAR := abreparenteses EXP_CHAR fechaparenteses
 | char:a mais char:b
 | char:a menos char:b
 | char

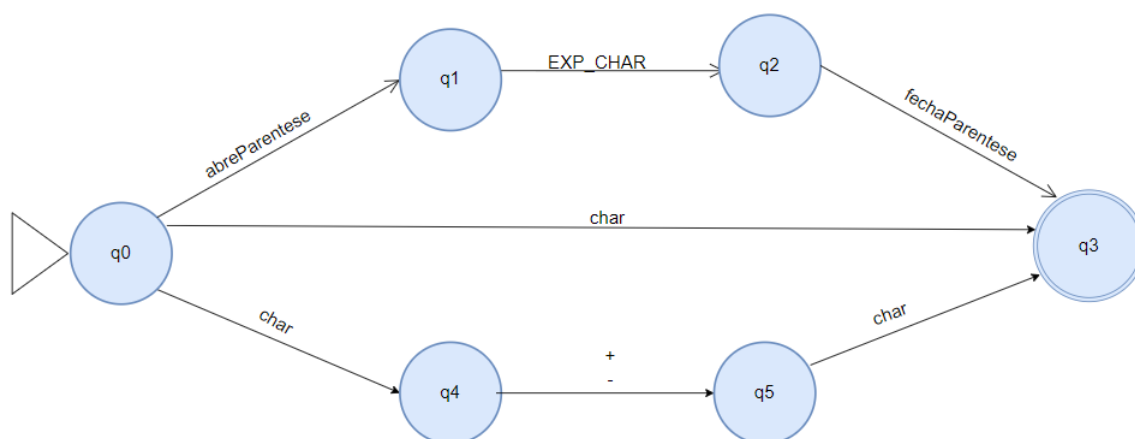


Figura 44 – Autômato - EXP_CHAR

3.2 Entradas e saídas

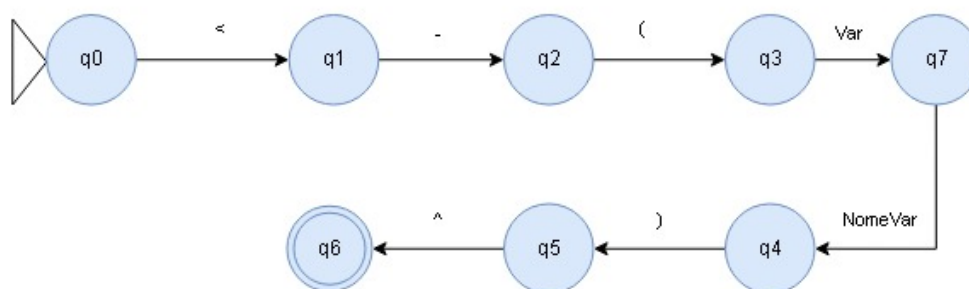


Figura 45 – Autômato relacionado à saída

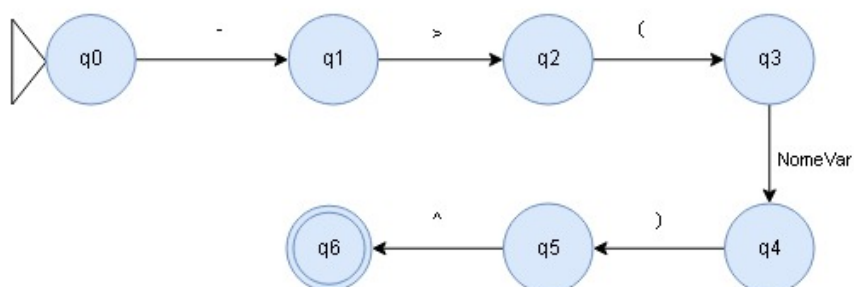


Figura 46 – Autômato relacionado à entrada

3.3 Expressões Relacionais

Em relação a abstração das operações relacionais podemos ter:

OP REL := | igual

| maiorigual

| menorigual

| maior

| menor

| diferente Enquanto que para as expressões relacionais , temos as regras que

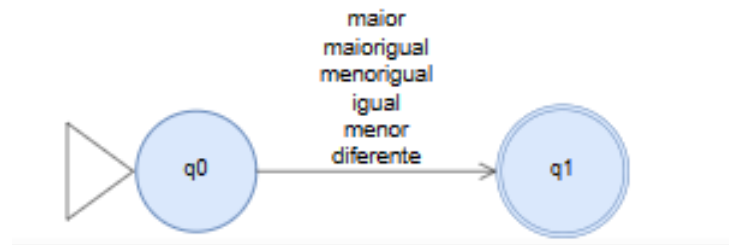


Figura 47 – Autômato - Operadores Relacionais

também envolvem operadores relacionais.

EXP REL := abreparenteses EXP REL fechaparenteses

| id op_rel id

| nomeVar op_rel nomeVar

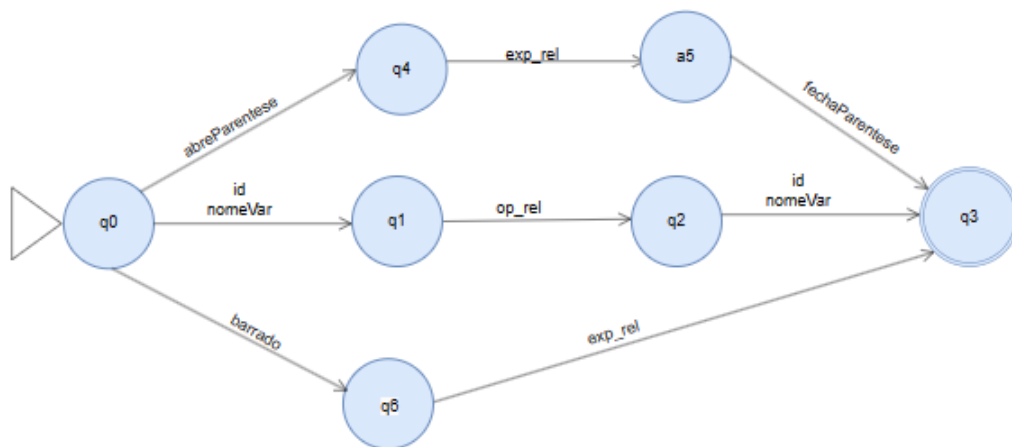


Figura 48 – Autômato - Expressões Relacionais

3.4 Expressões Lógicas

Para os operadores lógicos temos as seguintes opções:

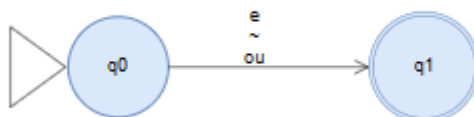


Figura 49 – Autômato - Operadores Lógicos

Enquanto que para as expressões relacionais, temos as regras que também envolvem operadores relacionais.

EXP_LOGICA := abreparenteses EXP_LOGICA fechaparenteses

| id op_logico id

| nomeVar op_logico nomeVar

| barrado exp_logica

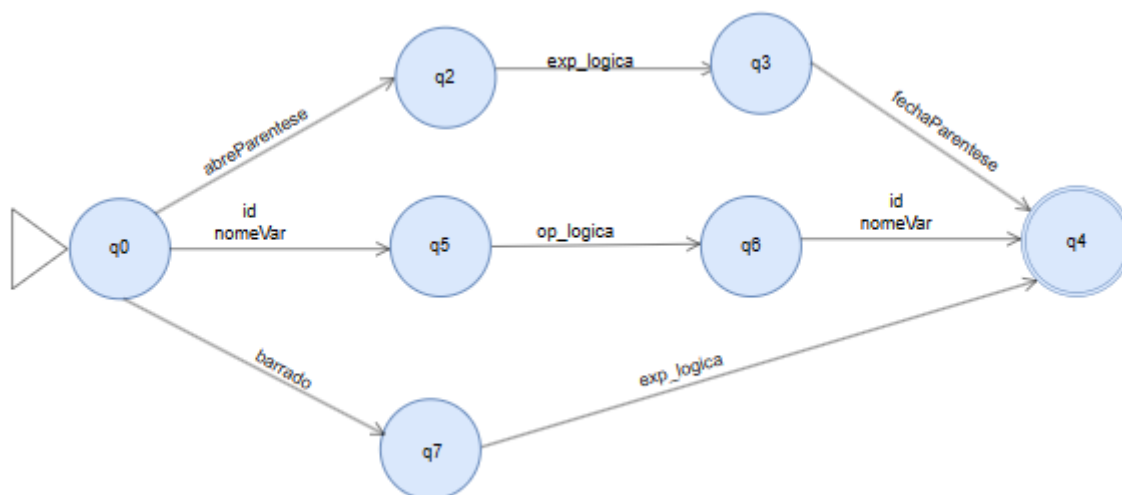


Figura 50 – Autômato - Expressões Lógicas

3.5 Declarando Variáveis

Quando se diz respeito a declaração de variáveis podemos ter por exemplo "intero nomeVar circunflexo" ou "intero nomeVar = id circunflexo".

VAR := intero|real|bool|caractere NomeVar circunflexo

| intero|real|bool|caractere NomeVar atribuicao id circunflexo

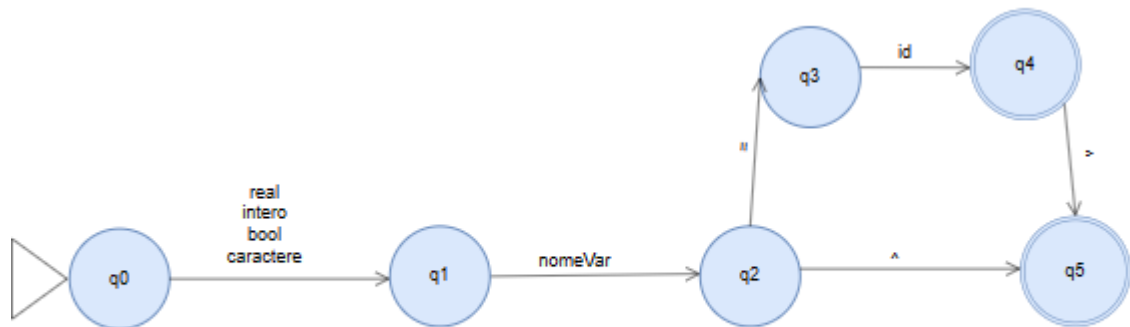


Figura 51 – Autômato - retorna

3.6 Estruturas Condicionais

Nessa seção se aborda as estruturas condicionais, onde o autômato relacionado funciona com base nas seguintes operações condicionais:

CASO := se abrepar EXPLOGICA fechar comeco BLOCO fim

NAODEU := senao comeco BLOCO fim

ESE := senaose abrepar EXPLOGICA fechar comeco BLOCO fim

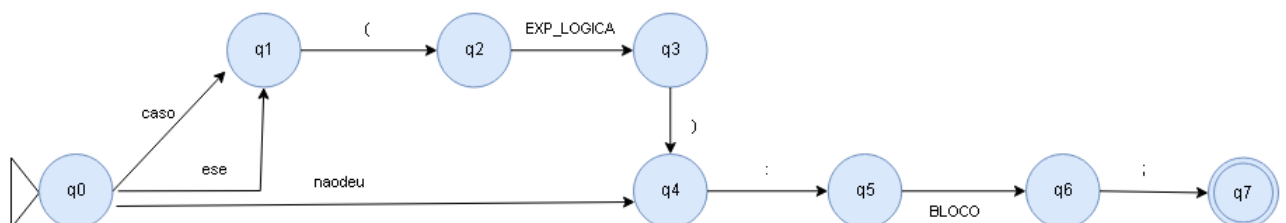


Figura 52 – Autômato das Estruturas Condicionais

3.7 Funções

O autômato abaixo se refere à criação de funções na linguagem, existindo também os formatos de criação a seguir:

FUNC := VAR NomeVar abrepar VAR NomeVar fechapar comeco BLOCO retorna Nomevar fimcomando fim

| VAR NomeVar abrepar VAR NomeVar fechapar comeco BLOCO fim

| VAR NomeVar abrepar fechapar comeco BLOCO retorna Nomevar fimcomando fim

| VAR NomeVar abrepar fechapar comeco BLOCO retorna Nomevar fimcomando fim

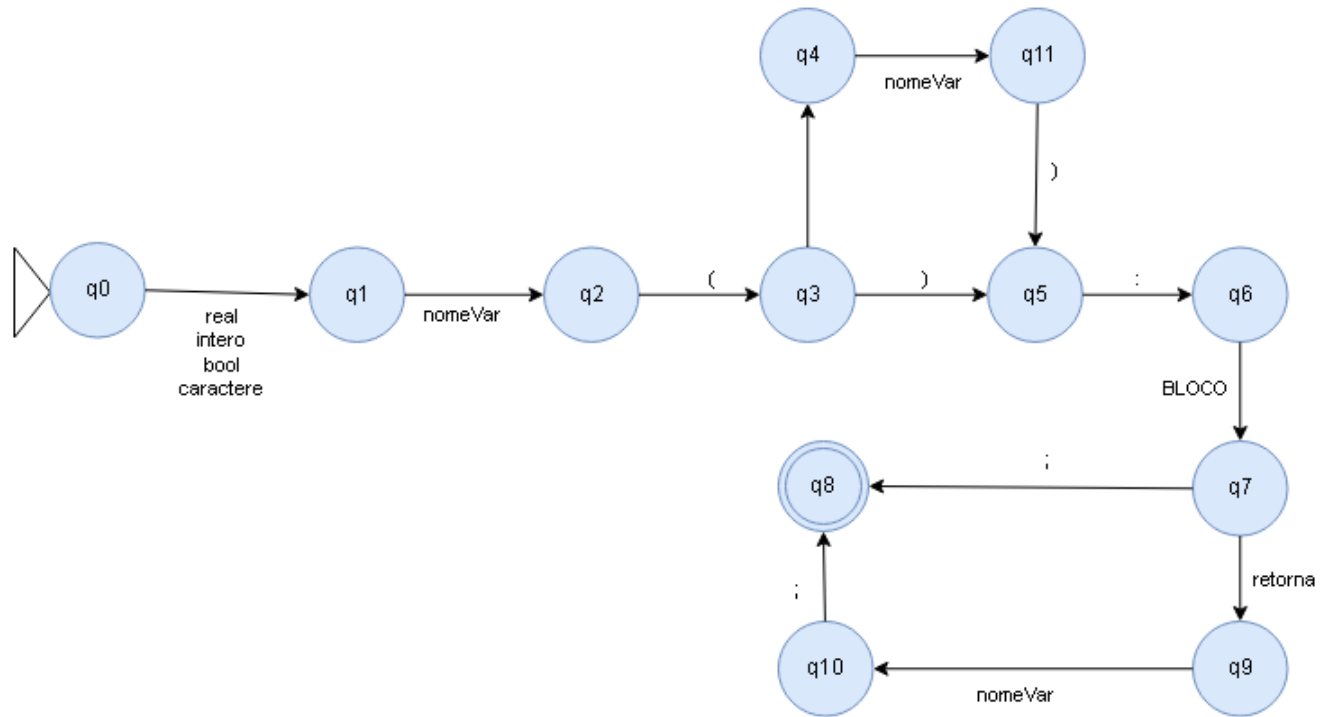


Figura 53 – Autômato de Criação de Funções

Já a próxima produção faz referência à chamada de funções na linguagem:

CFUNC := VAR NomeVar abrepar fechapar fim

| VAR NomeVar abrepar VAR NomeVar fechapar fim

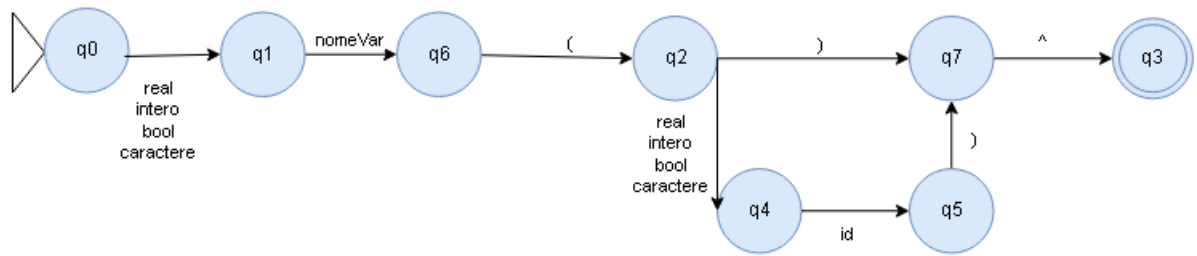


Figura 54 – Autômato de Chamada de Funções

3.8 Estruturas de Repetição

O autômato abaixo contém as estruturas de repetição "uai" e "floop", com cada uma delas apresentando a seguinte estrutura:

UAI := enquanto abrepar EXP_LOGICA fecharpar começo BLOCO fim

FLOOP := para NomeVar no abrepar int fecharpar começo BLOCO fim

| NomeVar no abrepar int virgula int fecharpar começo BLOCO fim

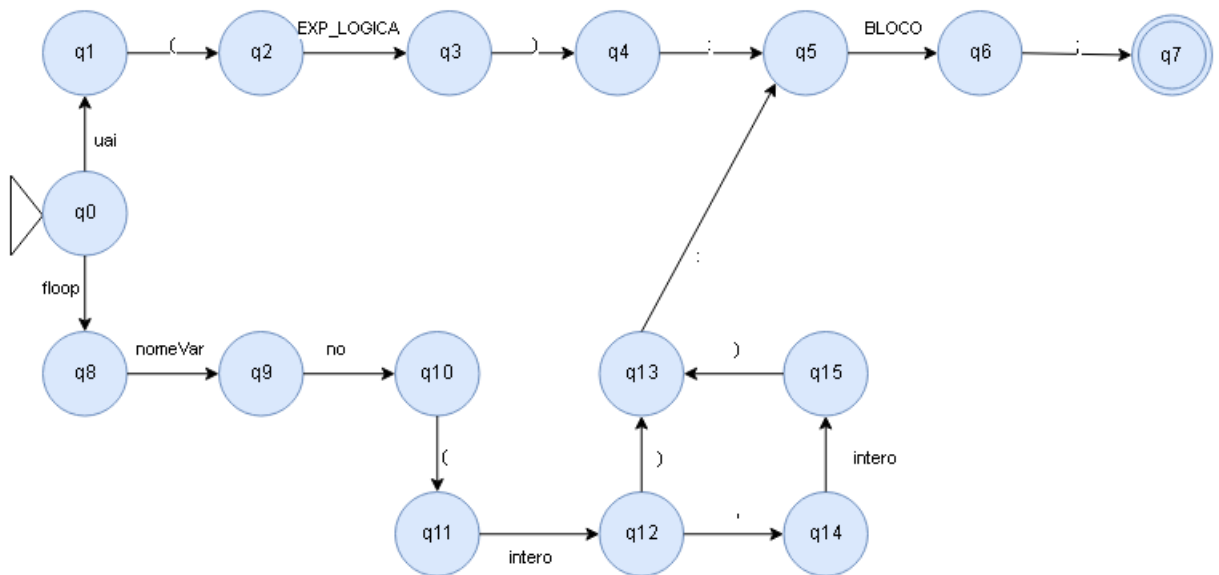


Figura 55 – Autômato de Estruturas de Repetição