



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

Instituto de Ciências Exatas e de Informática

Departamento de Ciência da Computação

Disciplina: Compiladores

Compilador para a linguagem de programação L

*

Vinícius Francisco da Silva¹

*Trabalho apresentado para a disciplina de compiladores.

¹Aluno do Programa de Graduação em Ciência da Computação, Brasil – vinicius.silva.1046664@sga.pucminas.br.

1 ALFABETO Σ

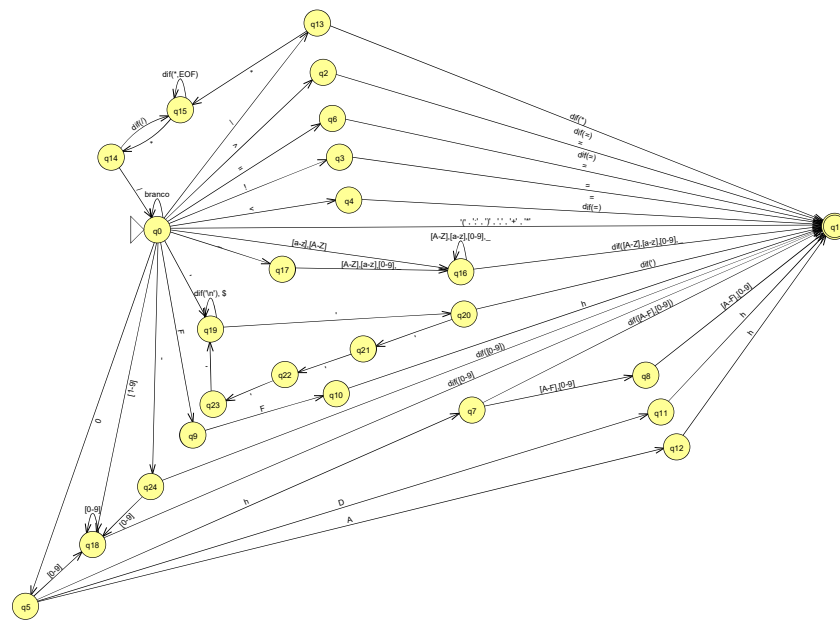
- const
- integer
- byte
- string
- while
- if
- else
- and
- or
- not
- =
- ==
- (
-)
- <
- >
- !=
- >=
- <=
- ,
- +
- -
- *
- /
- ;

- begin
- end
- then
- readln
- main
- writeln
- true
- false
- boolean
- id
- constante

2 PADRÃO DE FORMAÇÃO DOS LEXEMAS

Tabela 1 – Lexema x Padrão de formação

Posição	Lexema	Padrão de Formação
1	const	$(c \cup C)(o \cup O)(n \cup N)(s \cup S)(t \cup T)$
2	integer	$(i \cup I)(n \cup N)(t \cup T)(e \cup E)(g \cup G)(e \cup E)(r \cup R)$
3	byte	$(b \cup B)(y \cup Y)(t \cup T)(e \cup E)$
4	string	$(s \cup S)(t \cup T)(r \cup R)(i \cup I)(n \cup N)(g \cup G)$
5	while	$(w \cup W)(h \cup H)(i \cup I)(l \cup L)(e \cup E)$
6	if	$(i \cup I)(f \cup F)$
7	else	$(e \cup E)(l \cup L)(s \cup S)(e \cup E)$
8	and	$(a \cup A)(n \cup N)(d \cup D)$
9	or	$(o \cup O)(r \cup R)$
10	not	$(n \cup N)(o \cup O)(t \cup T)$
11	=	=
12	==	==
13	()
14)	(
15	<	<
16	>	>
17	!=	!=
18	>=	>=
19	<=	<=
20	,	,
21	+	+
22	-	-
23	*	*
24	/	/
25	;	;
26	begin	$(b \cup B)(e \cup E)(g \cup G)(i \cup I)(n \cup N)$
27	end	$(e \cup E)(n \cup N)(d \cup D)$
28	then	$(t \cup T)(h \cup H)(e \cup E)(n \cup N)$
29	readln	$(r \cup R)(e \cup E)(a \cup A)(d \cup D)(l \cup L)(n \cup N)$
30	main	$(m \cup M)(a \cup A)(i \cup I)(n \cup N)$
31	writeln	$(w \cup W)(r \cup R)(i \cup I)(t \cup T)(e \cup E)(l \cup L)(n \cup N)$
32	true	$(t \cup T)(r \cup R)(u \cup U)(e \cup E)$
33	false	$(f \cup F)(a \cup A)(l \cup L)(s \cup S)(e \cup E)$
34	boolean	$(b \cup B)(o \cup O)(o \cup O)(l \cup L)(e \cup E)(a \cup A)(n \cup N)$
35	id	$(L \cup l)(L \cup l \cup d \cup _)* \cup (_)(L \cup l \cup d \cup _)^+$
36	constante	d^+



GRAMÁTICA

S → {**Declarar**}* | “main” {**Declarar**}* {**Comandos**}* “end”

Escrever → “write” (“**Expressao**”) “;” | “writeln” (“**Expressao**”) “;”

Ler → “readln” (“**id**”) “;”

Atribuir → “id” “=” **Expressao** “;”

ListaAtribuir → (“integer”|“byte”|“boolean”|“string”) “id” {“,” “id” “=” “valor”}* “;”

Declarar → “const” “id” “=” “valor” “;” | **ListaAtribuir**

Desvio → “if” (“**Expressao**”) “then” **Comando** | “if” (“**Expressao**”) “then” **Comando** “else”

Comando | “if” (“**Expressao**”) “then” “begin” {**Comandos**}* “end” “else” “begin” {**Comandos**}* “end”

Repetir → “while” (“**Expressao**”) **Comando** | “while” (“**Expressao**”)+ “begin” {**Comando**}* “end”

Comando → **Ler** | **Escrever** | **Desvio** | **Atribuir** | **Repetir**

Expressao → “id” | {**Relacional** | **Aritmetico** | **Logico**}+ | (“**Expressao**”)

Relacional → **Expressao**[(“==” | “!=” | “<” | “>” | “<=” | “>=”) **Expressao**]

Aritmetico → **Expressao**[(“+” | “*” | “/” | “-”) **Expressao**]

Logico → “not” **Expressao** | **Expressao**[(“or” | “and”) **Expressao**]