



Centro Federal de Educação Tecnológica de Minas Gerais

Compiladores - Trabalho Prático

Parte 1 - Analisador Léxico e Tabela de Símbolos

Membros: Juliano Martins
Mariana Athayde
Thais Diniz Braz

Belo Horizonte / MG
2015

Implementação

Nessa primeira parte do trabalho, foram implementados o Analisador Léxico e a Tabela de Símbolos. Trata-se do início da implementação de um compilador.

O objetivo do analisador léxico é ler o caracteres do programa fonte, agrupá-los em lexemas, identificar o token correspondente ao lexema e gerar com saída uma sequência de tokens, os quais serão entrada para o analisador sintático, a ser implementado na próxima etapa deste trabalho.

Para essa implementação, foi adotado a seguinte forma de um token:

par<nome, valor>

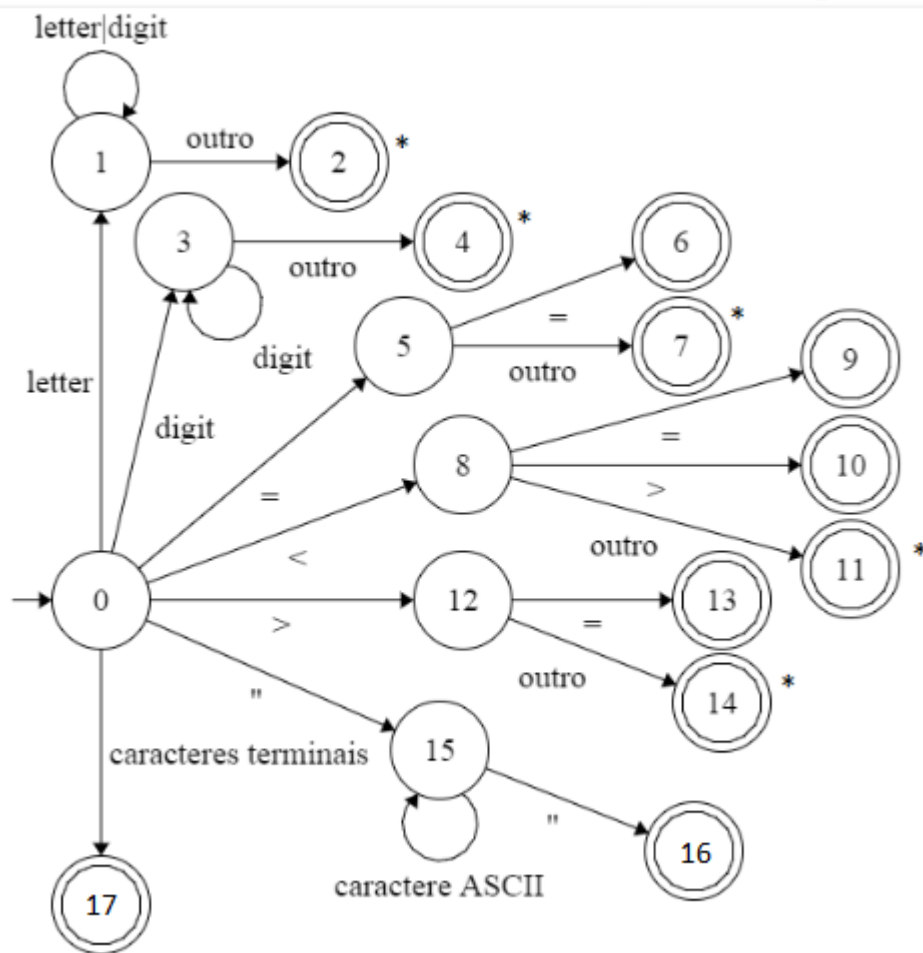
Onde nome é um símbolo abstrato que representa um tipo de unidade léxica (por exemplo, identificador, palavras reservadas, constantes, etc) e valor é a descrição do lexema representado pelo token.

A codificação adotada para o campo valor é a que segue:

Valor	Tipo do Token
1	DCL: Palavra reservada declare
2	STRT: Palavra reservada start
3	END: Palavra reservada end
4	INT: Palavra reservada int
5	STRG: Palavra reservada string
6	IF: Palavra reservada if
7	THEN: Palavra reservada then
8	ELSE: Palavra reservada else
9	DO: Palavra reservada do
10	WHL: Palavra reservada while
11	READ: Palavra reservada read
12	WRT: Palavra reservada write
13	OR: Palavra reservada or
14	AND: Palavra reservada and
20	EQ: Símbolo de igual (==)
21	GT: Símbolo de maior (>)
22	GE: Símbolo de maior ou igual (>=)

23	LT: Símbolo de menor (<)
24	LE: Símbolo de maior ou igual (<=)
25	NE: Símbolo de diferente (<>)
26	SUM: Sinal de soma (+)
27	MIN: Sinal de subtração (-)
28	MUL: Sinal de multiplicação (*)
29	DIV: Símbolo de divisão (/)
30	PVG: Ponto e vírgula (;)
31	VG: Vírgula (,)
32	DTPS: Dois pontos(:)
33	AP: Abre parênteses (())
34	FP: Fecha parênteses ())
35	ATBR: Símbolo de atribuição (=)
36	LIT: Representa um literal
40	NUM: Representa um número
41	ID: Representa um identificador
42	NESP: Não especificado

A implementação deste Analisador Léxico foi baseado no AFD, definido a partir de uma gramática pré determinada, representado na imagem abaixo.



Neste diagrama, por motivos de simplificação, todos os símbolos terminais (+ , - , (,) , ; , * , , , /) foram considerados pelo estado 17, representando que: partindo do estado inicial, qualquer um destes caracteres levam ao seu próximo estado, sendo este final, não dependendo ou estando relacionados a qualquer outros caracter.

Errata: Os comentários, não representados no diagrama, são reconhecidos semelhantemente aos literais (estados 15 e 16), porém quando o analisador reconhece os lexemas “//” e “/*” (estados 0 -> 15) os próximos caracteres são ingorados, até que o analisador reconheça o ponto quer representam o encerramento deste estado (estado 15 -> 16, “\n” e “*/” respectivamente).

A tabela baixo relaciona os estados a cada representação:

Estado	Representa
2	Identificador
4	Números inteiros
6	Comparação: igual a
7	Atribuição

9	Comparação: menor ou igual a
10	Diferença <>
11	Comparação: Menor que
12	Comparação: maior ou igual a
14	Comparação: Maior que
16	Literal
17	,
17	;
17	+
17	-
17	*
17	/
17	(
17)

Os principais aspectos a serem ressaltados em nosso código são:

- **A classe `Lexer.java`**

Esta classe lê o arquivo de entrada, caractere a caractere, e retorna os tokens encontrados.

- **As estruturas `TabelaSimbolos` e `TabelaReservado`**

Ambas tabelas são estruturadas usando `HashMap`, e contêm, respectivamente, todos os identificadores encontrados no programa e todas palavras reservadas pré definidas pela gramática, ao qual este analisador implementa, as quais são “instaladas” no início da execução do programa.

Além dos pontos acima, as demais classes servem de auxílio às principais: A classe `TAG.java` apenas declara e instância os valores inteiros que representam qual o tipo do token utilizado, a classe `Token.java` representa os tokens que serão reconhecidos, a classe `Word` estende a classe `Token` e refere-se ao identificadores, assim como a classe `Num` que refere-se aos números.

Forma de uso do compilador

Para utilizar o compilador deve-se chamar o arquivo compilador.jar, passando como parâmetro os arquivos que serão compilados. Lembrando que nessa primeira parte do trabalho, compilar consiste apenas em imprimir os Tokens reconhecidos (ou os erros para os não reconhecidos) e as entradas da Tabela de Símbolos.

Exemplo:

O comando 'java -jar compilador.jar teste1' compila o arquivo teste1.

Testes

Nesta etapa foram realizados os testes no analisador léxico, para tal foram considerados 8 arquivos, dentre eles 7 contendo erros léxicos. Abaixo estarão as evidências de execução de cada um dos testes antes e depois da correção dos erros, é importante ressaltar que aqui não são corrigidos os outros erros que não dizem respeito ao analisador léxico (erro sintático/semântico).

Arquivo de Teste 1

Esse arquivo contém um erro léxico que consiste em um caracter não contido no alfabeto da gramática.

Programa: teste1 program teste1 declare a, b, c: int; result: int start read (a); read (c); b = 10; result = (a * c)/(b + 5 - 345); write(result); teste1 = 1; end	<),34> <,,30> <b,41> <=,35> <10,40> <,,30> <result,41> <=,35> <(,33> <a,41> <*,28> <c,41> <),34> </,29> <(,33>
Resultado do Teste 1: <program,41> <teste1,41> <declare,41> <a,41> <,,31> <b,41> <,,31> <c,41> <:,32> <int,41> <,,30>	<b,41> <+,26> <5,40> <-,27> <345,40> <),34> <,,30> <write,41> <(,33> <result,41> <),34> <,,30> <teste1,41>

<result,41> <:,32> <int,41> <start,41> <read,41> <(,33> <a,41> <),34> <:,30> <read,41> <(,33> <c,41>	<=,35> <1,40> <:,30> <end,41> <----- TABELA DE SÍMBOLOS -----> result a b c teste1 program CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
---	---

Arquivo de Teste 2

Esse arquivo contém um erro léxico que consiste em um caracter não contido no alfabeto da gramática.

Programa: teste2 declare a: int; b: int; nome: string; start read (a); read (nome); b = a * a; b = b + a/2 * (3 + 5); write(nome); write(b); end.	Programa sem erros léxicos: teste2_Corrigido declare a: int; b: int; nome: string; start read (a); read (nome); b = a * a; b = b + a/2 * (3 + 5); write(nome); write(b); end
Resultado do Teste 2: <declare,41> <a,41> <:,32> <int,41> <:,30> <b,41> <:,32> <int,41> <:,30> <nome,41> <:,32> <string,41> <:,30> <start,41> <read,41> <(,33> <a,41>	Resultado do Teste 2 Corrigido: <declare,41> <a,41> <:,32> <int,41> <:,30> <b,41> <:,32> <int,41> <:,30> <nome,41> <:,32> <string,41> <:,30> <start,41> <read,41> <(,33> <a,41>

<),34> <;,30> <read,41> <(,33> <nome,41> <),34> <;,30> <b,41> <=,35> <a,41> <*,28> <a,41> <;,30> <b,41> <=,35> <b,41> <+,26> <a,41> </,29> <2,40> <*,28> <(,33> <3,40> <+,26> <5,40> <),34> <;,30> <write,41> <(,33> <nome,41> <),34> <;,30> <write,41> <(,33> <b,41> <),34> <;,30> <end,41> ERRO - Caractere não especificado na linha 12 ch: ".". <.,42> <----- TABELA DE SÍMBOLOS -----> a b nome CONSTRUÍDO COM SUCESSO (tempo total: 2 segundos)	<),34> <;,30> <read,41> <(,33> <nome,41> <),34> <;,30> <b,41> <=,35> <a,41> <*,28> <a,41> <;,30> <b,41> <=,35> <b,41> <+,26> <a,41> </,29> <2,40> <*,28> <(,33> <3,40> <+,26> <5,40> <),34> <;,30> <write,41> <(,33> <nome,41> <),34> <;,30> <write,41> <(,33> <b,41> <),34> <;,30> <end,41> <----- TABELA DE SÍMBOLOS -----> a b nome CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
---	---

Arquivo de Teste 3

Esse arquivo contém um erro léxico que consiste em um caracter não contido no alfabeto da gramática.

Programa: teste3 declare cont: int; media, altura, soma: int; start cont = 5; soma = 0; do write("Altura: "); read (altura); soma = soma # altura; cont = cont - 1; while(cont); write("Media: "); write (soma / qtd); end	Programa sem erros léxicos: teste3_Corrigido declare cont: int; media, altura, soma: int; start cont = 5; soma = 0; do write("Altura: "); read (altura); soma = soma * altura; cont = cont - 1; while(cont); write("Media: "); write (soma / qtd); end
Resultado do Teste 3: <declare,41> <cont,41> <:,32> <int,41> <:,30> <media,41> <,,31> <altura,41> <,,31> <soma,41> <:,32> <int,41> <:,30> <start,41> <cont,41> <=,35> <5,40> <:,30> <soma,41> <=,35> <0,40> <:,30> <do,41> <write,41>	Resultado do Teste 3 Corrigido: <declare,41> <cont,41> <:,32> <int,41> <:,30> <media,41> <,,31> <altura,41> <,,31> <soma,41> <:,32> <int,41> <:,30> <start,41> <cont,41> <=,35> <5,40> <:,30> <soma,41> <=,35> <0,40> <:,30> <do,41> <write,41>

<pre> <(,33> <"Altura: ",36> <),34> <;,30> <read,41> <(,33> <altura,41> <),34> <;,30> <soma,41> <=,35> <soma,41> ERRO - Caractere não especificado na linha 10 ch: "#". <#,42> <altura,41> <;,30> <cont,41> <=,35> <cont,41> <-,27> <1,40> <;,30> <while,41> <(,33> <cont,41> <),34> <;,30> <write,41> <(,33> <"Media: ",36> <),34> <;,30> <write,41> <(,33> <soma,41> </,29> <qtd,41> <),34> <;,30> <end,41> <----- TABELA DE SÍMBOLOS -----> qtd altura soma media cont </pre>	<pre> <(,33> <"Altura: ",36> <),34> <;,30> <read,41> <(,33> <altura,41> <),34> <;,30> <soma,41> <=,35> <soma,41> <*,28> <altura,41> <;,30> <cont,41> <=,35> <cont,41> <-,27> <1,40> <;,30> <while,41> <(,33> <cont,41> <),34> <;,30> <write,41> <(,33> <"Media: ",36> <),34> <;,30> <write,41> <(,33> <soma,41> </,29> <qtd,41> <),34> <;,30> <----- TABELA DE SÍMBOLOS -----> qtd altura soma media cont CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos) </pre>
--	---

Arquivo de Teste 4

Esse arquivo contém três erros léxicos que consiste em identificador mal formado (inicializados de um número) e caracteres não especificados no alfabeto da gramática.

<p>Programa: teste4</p> <pre> declare i, j, k, @total, 1soma: int; start i = 4 * (5-3 * 50 / 10; j = i * 10; k = i* j / k; k = 4 + a \$; write(i); write(j); write(k); end </pre>	<p>Programa sem erros léxicos: teste4_Corrigido</p> <pre> declare i, j, k, total, soma1: int; start i = 4 * (5-3 * 50 / 10; j = i * 10; k = i* j / k; k = 4 + a; write(i); write(j); write(k); end </pre>
<p>Resultado do Teste 4:</p> <pre> <declare,41> <i,41> <,,31> <j,41> <,,31> <k,41> <,,31> ERRO - Caractere não especificado na linha 2 ch: "@". <@,42> <total,41> <,,31> Erro! Identificador incorreto na linha 2 <1,40> <soma,41> <:,32> <int,41> <;,30> <start,41> <i,41> <=,35> <4,40> <*,28> <(,33> <5,40> <-,27> </pre>	<p>Resultado do Teste 4 Corrigido:</p> <pre> <declare,41> <i,41> <,,31> <j,41> <,,31> <k,41> <,,31> <total,41> <,,31> <soma1,41> <:,32> <int,41> <;,30> <start,41> <i,41> <=,35> <4,40> <*,28> <(,33> <5,40> <-,27> <3,40> <*,28> <50,40> </,29> <10,40> </pre>

<3,40> <*,28> <50,40> </,29> <10,40> <;,30> <j,41> <=,35> <i,41> <*,28> <10,40> <;,30> <k,41> <=,35> <i,41> <*,28> <j,41> </,29> <k,41> <;,30> <k,41> <=,35> <4,40> <+,26> <a,41> ERRO - Caractere não especificado na linha 7 ch: "\$". <\$,42> <;,30> <write,41> <(,33> <i,41> <),34> <;,30> <write,41> <(,33> <j,41> <),34> <;,30> <write,41> <(,33> <k,41> <),34> <;,30> <end,41> <----- TABELA DE SÍMBOLOS -----> a total soma i j	<;,30> <j,41> <=,35> <i,41> <*,28> <10,40> <;,30> <k,41> <=,35> <i,41> <*,28> <j,41> </,29> <k,41> <;,30> <k,41> <=,35> <4,40> <+,26> <a,41> <;,30> <write,41> <(,33> <i,41> <),34> <;,30> <write,41> <(,33> <j,41> <),34> <;,30> <write,41> <(,33> <k,41> <),34> <;,30> <end,41> <----- TABELA DE SÍMBOLOS -----> a total i soma1 j k CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
---	---

k CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)	
--	--

Arquivo de Teste 5

Esse arquivo contém um erro onde o comentário é iniciado entretanto, não é encerrado.

Programa: teste5 <pre> declare /*Este e um programa com if j, k: int; a, j: real; start read(j); read(k); if (k != 0) result = j/k else result = 0 end; write(res); end </pre>	Programa sem erros léxicos: teste5_Corrigido <pre> declare /*Este e um programa com*/ if j, k: int; a, j: real; start read(j); read(k); if (k == 0) result = j/k else result = 0 end; write(res); end </pre>
Resultado do Teste 5: <pre> <declare,41> WARNING! Comentário não encerrado! <----- TABELA DE SÍMBOLOS -----> CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo) </pre>	Resultado do Teste 5 Corrigido: <pre> <declare,41> <if,41> <j,41> <,,31> <k,41> <:,32> <int,41> <,,30> <a,41> <,,31> <j,41> <:,32> <real,41> <,,30> <start,41> <read,41> <(,33> <j,41> <),34> <,,30> <read,41> <(,33> <k,41> <),34> </pre>

	<,,30> <if,41> <(,33> <k,41> <==,20> <0,40> <),34> <result,41> <=,35> <j,41> </,29> <k,41> <else,41> <result,41> <=,35> <0,40> <end,41> <,,30> <write,41> <(,33> <res,41> <),34> <,,30> <end,41> <----- TABELA DE SÍMBOLOS -----> result a res j k real
--	---

Arquivo de Teste 6

Esse arquivo contém dois erros léxicos: possui identificador maior do que o tamanho máximo permitido e extrapola o limite de tamanho do tipo inteiro.

Programa: teste6 declare // Um programa para calcular a maior idade a, b, c, maior: int; umavariavelcomonomemuitogrande: string; start read(a); read(b); read(c); maior = 12345678901234567890123456789; if (a>b and a>c) then maior = a; else if (b>c) then maior = b;	Programa sem erros léxicos: teste6_Corrigido declare // Um programa para calcular a maior idade a, b, c, maior: int; variavel: string; start read(a); read(b); read(c); maior = 123456; if (a>b and a>c) then maior = a; else if (b>c) then
--	---

<pre> else maior = c; end end write("Maior idade: "); write(maior); end end </pre>	<pre> maior = b; else maior = c; end end write("Maior idade: "); write(maior); end end </pre>
<p>Resultado do Teste 6:</p> <pre> <declare,41> <a,41> <,,31> <b,41> <,,31> <c,41> <,,31> <maior,41> <:,32> <int,41> <;,30> ERRO! Identificador excedeu o número permitido de caracteres! <umavariavelcomonomem,41> <uitogrande,41> <:,32> <string,41> <;,30> <start,41> <read,41> <(,33> <a,41> <),34> <;,30> <read,41> <(,33> <b,41> <),34> <;,30> <read,41> <(,33> <c,41> <),34> <;,30> <maior,41> <=,35> Erro! Valor de inteiro maior que o maximo permitido <12345678901234567890123456789,40> <;,30> </pre>	<p>Resultado do Teste 6 Corrigido:</p> <pre> <declare,41> <a,41> <,,31> <b,41> <,,31> <c,41> <,,31> <maior,41> <:,32> <int,41> <;,30> <variavel,41> <:,32> <string,41> <;,30> <start,41> <read,41> <(,33> <a,41> <),34> <;,30> <read,41> <(,33> <b,41> <),34> <;,30> <read,41> <(,33> <c,41> <),34> <;,30> <maior,41> <=,35> <123456,40> <;,30> <if,41> <(,33> <a,41> <>,21> <b,41> </pre>

<if,41> <(,33> <a,41> <>,21> <b,41> <and,41> <a,41> <>,21> <c,41> <),34> <then,41> <maior,41> <=,35> <a,41> <;,30> <else,41> <if,41> <(,33> <b,41> <>,21> <c,41> <),34> <then,41> <maior,41> <=,35> <b,41> <;,30> <else,41> <maior,41> <=,35> <c,41> <;,30> <end,41> <end,41> <write,41> <(,33> <"Maior idade: ",36> <),34> <;,30> <write,41> <(,33> <maior,41> <),34> <;,30> <end,41> <end,41> <----- TABELA DE SÍMBOLOS -----> a b c maior	<and,41> <a,41> <>,21> <c,41> <),34> <then,41> <maior,41> <=,35> <a,41> <;,30> <else,41> <if,41> <(,33> <b,41> <>,21> <c,41> <),34> <then,41> <maior,41> <=,35> <b,41> <;,30> <else,41> <maior,41> <=,35> <c,41> <;,30> <end,41> <end,41> <write,41> <(,33> <"Maior idade: ",36> <),34> <;,30> <write,41> <(,33> <maior,41> <),34> <;,30> <end,41> <end,41> <----- TABELA DE SÍMBOLOS -----> a b c maior variavel CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)
--	--

umavariavelcomonomem uitogrande CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)	
--	--

Arquivo de Teste 7

Esse arquivo contém um erro léxico que consiste em um termo literal que é inicializado entretanto, não é finalizado.

Programa: teste7 declare i, j, k, declare: int; start i = 4 ; j = *;(((tt=2; write(j); write(j)****/ write("Altura é /* & demais); k = i* j / k; k = 4 + a \$; write(j); write(k*); end; end	Programa sem erros léxicos: teste7_Corrigido declare i, j, k, declare: int; start i = 4 ; j = *;(((tt=2; write(j); write(j)****/ write("Altura é &demais"); k = i* j / k; k = 4 + a; write(j); write(k*); end; end
Resultado do Teste 7: <declare,41> <i,41> <,,31> <j,41> <,,31> <k,41> <,,31> <declare,41> <:,32> <int,41> <:,30> <start,41> <i,41> <=,35> <4,40> <:,30> <j,41> <=,35> <*,28> <:,30> <(,33> <(,33> <(,33> <*,28>	Resultado do Teste7 Corrigido: <declare,41> <i,41> <,,31> <j,41> <,,31> <k,41> <,,31> <declare,41> <:,32> <int,41> <:,30> <start,41> <i,41> <=,35> <4,40> <:,30> <j,41> <=,35> <*,28> <:,30> <(,33> <(,33> <(,33> <*,28>

<tt,41> <=,35> <2,40> <;,30> <write,41> <(,33> <j,41> <),34> <;,30> <write,41> <(,33> <j,41> <),34> <*,28> <*,28> <*,28> <*,28> </,29> <write,41> <(,33> ERRO! String literal não finalizada! <----- TABELA DE SÍMBOLOS -----> tt i j k	<tt,41> <=,35> <2,40> <;,30> <write,41> <(,33> <j,41> <),34> <;,30> <write,41> <(,33> <j,41> <),34> <*,28> <*,28> <*,28> <*,28> </,29> <write,41> <(,33> <"Altura é &demais",36> <),34> <;,30> <k,41> <=,35> <i,41> <*,28> <j,41> </,29> <k,41> <;,30> <k,41> <=,35> <4,40> <+,26> <a,41> <;,30> <write,41> <(,33> <j,41> <),34> <;,30> <write,41> <(,33> <k,41> <*,28> <),34> <;,30> <end,41> <;,30> <end,41>
---	--

	<p><----- TABELA DE SÍMBOLOS -----></p> <p>tt</p> <p>a</p> <p>i</p> <p>j</p> <p>k</p> <p>CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)</p>
--	--

Arquivo de Teste 8

Esse arquivo contém alguns erros léxicos e foram inseridos caracteres de formatação (\t) para verificar o comportamento do analisador léxico.

<p>Programa: teste8</p> <pre> programa teste8 declare a, b, c: int; result: int; start read [a]; read (b); c = -30; a = 15,7; wr ite (a+c); write (" sucesso!"); end </pre>	<p>Programa sem erros léxicos: teste8_Corrigido</p> <pre> programa teste8 declare a,b,c: int; result: int; start read a; read (b); c = -30; a = 15,7; wr ite (a+c); write (" sucesso!"); end </pre>
<p>Resultado do Teste 8:</p> <pre> <programa,41> <teste8,41> <declare,41> <a,41> <,,31> <b,41> <,,31> <c,41> <:,32> <int,41> <:,30> <result,41> <:,32> <int,41> <:,30> <start,41> <read,41> ERRO - Caractere não especificado na linha 6 ch: "[". <[,42> </pre>	<p>Resultado do Teste 8 Corrigido:</p> <pre> <programa,41> <teste8,41> <declare,41> <a,41> <,,31> <b,41> <,,31> <c,41> <:,32> <int,41> <:,30> <result,41> <:,32> <int,41> <:,30> <start,41> <read,41> <a,41> <:,30> <read,41> </pre>

<a,41> ERRO - Caractere não especificado na linha 6 ch: "]" . <],42> <;,30> <read,41> <(,33> <b,41> <),34> <;,30> <c,41> <=,35> <-,27> <30,40> <;,30> <a,41> <=,35> <15,40> <,,31> <7,40> <;,30> <wr,41> <ite,41> <(,33> <a,41> <+,26> <c,41> <),34> <;,30> <write,41> <(,33> ERRO! String literal não finalizada! <----- TABELA DE SÍMBOLOS -----> result a teste8 b c programa wr ite CONSTRUÍDO COM SUCESSO (tempo total: 1 segundo)	<(,33> <b,41> <),34> <;,30> <c,41> <=,35> <-,27> <30,40> <;,30> <a,41> <=,35> <15,40> <,,31> <7,40> <;,30> <wr,41> <ite,41> <(,33> <a,41> <+,26> <c,41> <),34> <;,30> <write,41> <(,33> <" sucesso!",36> <),34> <;,30> <end,41> <----- TABELA DE SÍMBOLOS -----> result a teste8 b c programa wr ite CONSTRUÍDO COM SUCESSO (tempo total: 0 segundos)
--	--