

Modelagem e implantação de um Analisador Léxico para linguagem Java

Introdução:

Este documento contém a modelagem utilizada para o desenvolvimento de um analisador léxico. Na modelagem do analisador, considerou-se um conjunto de palavras e caracteres especiais reservados da linguagem Java, para geração de tokens pelo analisador. Além das palavras reservadas o analisador também pode identificar variáveis, valores, comentários feitos pelo usuário ou textos que estão entre em aspas que são considerados como String pelo analisador.

Modelagem do Sistema:

1. Palavras chaves (Keywords):

Foram escolhidas algumas palavras reservadas da linguagem Java para criação do analisador, a tabela a seguir contém todas as palavras reservadas e os seus respectivos tokens, gerados após a análise.

Keyword	Token
int, double, String	KW_DECLARACAO
if, else, switch, case, default	KW_CONDICIONAL
while, for, Do	KW_REPETICAO
break, continue	KW_CTRL_REP
public, private, protected	KW_MOD_ACESSO
print	METODO_ESCREVE
package, import	KW_CTRL_PACK
static, class, abstract, extends, final, implements, void	KW_MOD_CVM
return	KW_RETORNO
main	METODO_ESP_MAIN

2. Caracteres especiais:

Também foram escolhidos alguns caracteres especiais utilizados na linguagem Java, a tabela a seguir contém esses caracteres e os seus respectivos tokens.

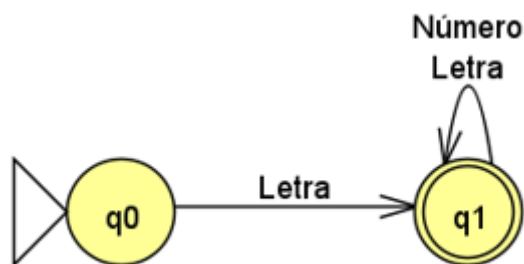
Caracter	Token
= <=	OPERADOR_REL
> ==	
< !	
>=	
()	DELIMITADOR
{ }	
[]	
* -	OPERADOR_MAT
+ /	
&&	OPERADOR_LOG
;	ENCERRA_COMANDO
,	SEPARADOR

3. Variáveis:

Para que o analisador identifique variáveis escritas no código do usuário foi utilizado a expressão regular a seguir que permite apenas palavras que comece com uma letra e após ela contenha apenas letras ou números.

$$^{\wedge}[a-zA-Z][a-zA-Z0-9]^{\ast}\$$$

A expressão pode ser representada pelo seguinte autômato:



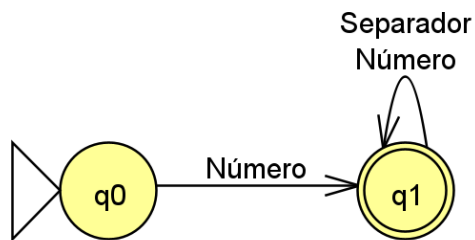
Onde Letra é igual a $[a-zA-Z]$ e número é $[0-9]$.

4. Valores:

Para que o analisador identifique valores escritos no código do usuário foi utilizado a expressão regular a seguir que permite apenas valores numéricos com ou sem um ponto entre eles.

$$^{\wedge}[0-9]^{\ast}(\.[0-9]^{\ast})? \$$$

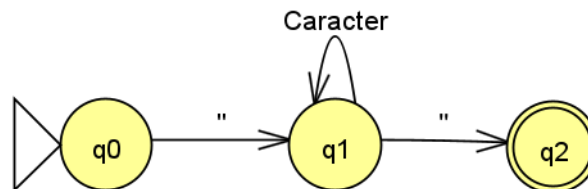
A expressão pode ser representada pelo seguinte autômato:



Onde Número é igual a [0-9] e Separador é igual a [.,].

5. Textos:

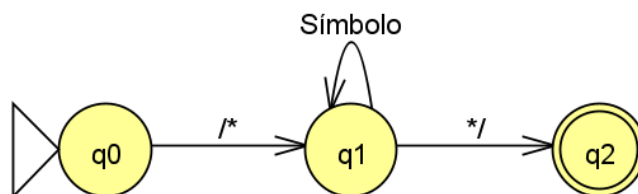
Para que o analisador identifique textos escritos no código do usuário foi utilizado a lógica do autômato a seguir:



Onde Caracter é qualquer caracter.

6. Comentários

Para que o analisador identifique comentários escritos no código do usuário foi utilizado a lógica do autômato a seguir:



Erros Léxicos:

1. Nomenclatura de variáveis

Caso a variável declarada pelo usuário não esteja de acordo com a expressão regular feita para as variáveis, o analisador retorna a mensagem de erro:

ERROR: Variável (nome da variável inválida) contém número ou caractere especial.

Exemplo de análise com variável definida com uma nomenclatura inválida:

Analizador Léxico

Arquivo:
C:\Users\andre\OneDrive\Área de Trabalho\Exe... 

Código:

```
int i# = 5;
while(i< 20){
    print("Hello world");
}
```

Tokens:

ERROR: Variável i# contém número ou caractere especial.

Pesquisa 

Documentação 

Validar 

2. Textos não finalizados

Caso o usuário tenha utilizado uma aspa dupla para limitar uma String e não tenha fechado as aspas duplas, o analisador retorna a mensagem de erro:

ERROR: String não finalizada.

Exemplo de análise com Textos não finalizados:

Analizador Léxico

Arquivo:
C:\Users\andre\OneDrive\Área de Trabalho\Exe... 

Código:

```
int i = 5;
while(i< 20){
    print("Hello world);
}
```

Tokens:

ERROR: String não finalizada.

Pesquisa 

Documentação 

Validar 

3. Comentários não finalizados

Caso o usuário tenha utilizado “/*” para iniciar um comentário e após esses dois símbolos não houver um “*/”, o analisador retorna a mensagem de erro:

ERROR: Comentário não finalizado.

Exemplo de análise com cometários não finalizados:

Analizador Léxico

Arquivo:
C:\Users\andre\OneDrive\Área de Trabalho\Exe...

Código:

```
int i = 5;
/*while (i< 20){
    print("Hello world");
}
```

Tokens:

ERROR: Comentário não finalizado.

Pesquisa Documentação Validar

Exemplo de análise sem erros Léxicos:

Analizador Léxico

Arquivo:
C:\Users\andre\OneDrive\Área de Trabalho\Exe...

Código:

```
int i = 5;
while (i< 20){
    /*print("Hello world");*/
}
```

Tokens:

KW_DECLARACAO	→	int
VARIABEL	→	i
OPERADOR_REL	→	=
VALOR	→	5
ENCERRA_COMANDO	→	;
KW_REPETICAO	→	while
DELIMITADOR	→	(
VARIABEL	→	i
OPERADOR_REL	→	<
VALOR	→	20
DELIMITADOR	→)

Pesquisa Documentação Validar

Grupo:

- Rafael Rossetto Guitarrari (RA: 823158602)
- Andrey de Freitas Souza (RA: 823217536)
- Gabriel Farah De Lima (RA: 822231424)
- Fabrício de Barros Narbon (RA: 822227166)
- Bianca Alves Ribeiro (RA: 8222240261)
- Luiz Gustavo França de Abreu (RA: 823210075)
- Gabrielle Garcia Paz (RA: 823126085)
- Webster Diógenes Rodrigues (RA: 8222242764)