Trabalho de Linguagens Formais, Autômatos e Computabilidade

Máquina de Turing

Rafael Francisco Ferreira

Ciência da Computação – Universidade Estadual do Paraná (UNESPAR) Apucarana – PR – Brasil

rafaelfrancisco_97@hotmail.com

1. Introdução

A Máquina de Turing é um dispositivo teórico conhecido como *máquina universal*, que foi concebido pelo matemático britânico Alan Turing (1912-1954), muitos anos antes de existirem os modernos computadores digitais. Num sentido preciso, é um modelo abstrato de um computador, que se restringe apenas aos aspectos lógicos do seu funcionamento (memória, estados e transições) e não à sua implementação física. Numa máquina de Turing pode-se modelar qualquer computador digital.

Uma máquina de Turing consiste em:

- 1. Fita: utilizada para leitura, rascunho e para escrita.
- 2. Cabeçote: mostra a posição atual da fita e se move para a direita e esquerda.
- 3. Função de Transição: função que movimenta a máquina a partir de um símbolo, gerando um novo símbolo, indo para um esta e movendo o cabeçote para a direita ou para a esquerda.
- 4. Representação Gráfica:



X: Símbolo lido da fita

Y: Símbolo escrito na fita

R: Sentido do movimento (Direita/Esquerda)

*A fita é limitada à esquerda pelo marcador de início e infinita à direita, contendo o símbolo branco ao final da palavra.

Tipos de MT:

- Reconhecedora: responde se uma palavra pertence ou não à linguagem.
- Transdutora: é gerada uma palavra na própria fita que é a saída da MT.

Este relatório apresenta uma implementação de uma Máquina de Turing Reconhecedora e Transdutora.

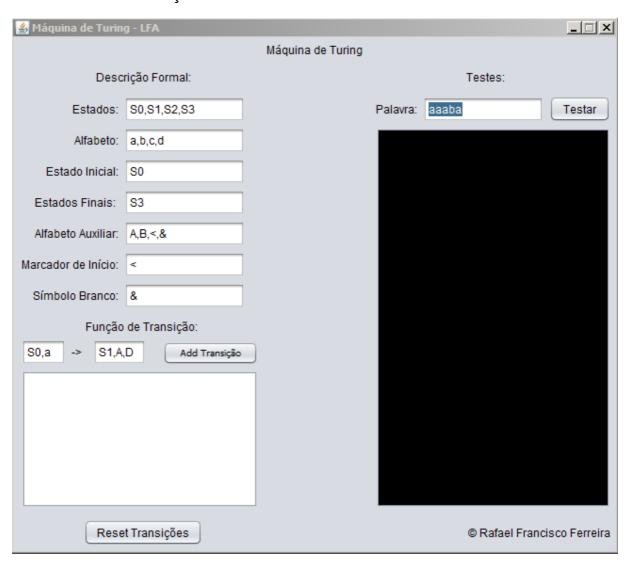
2. Objetivo

Esta implementação foi feita para um trabalho proposto na matéria de Linguagens Formais, Autômatos e Computabilidade.

A máquina tem a função de receber palavras e dizer se pertencem ou não à linguagem informada, ou mostrar as modificações feitas nas mesmas.

3. A Máquina de Turing

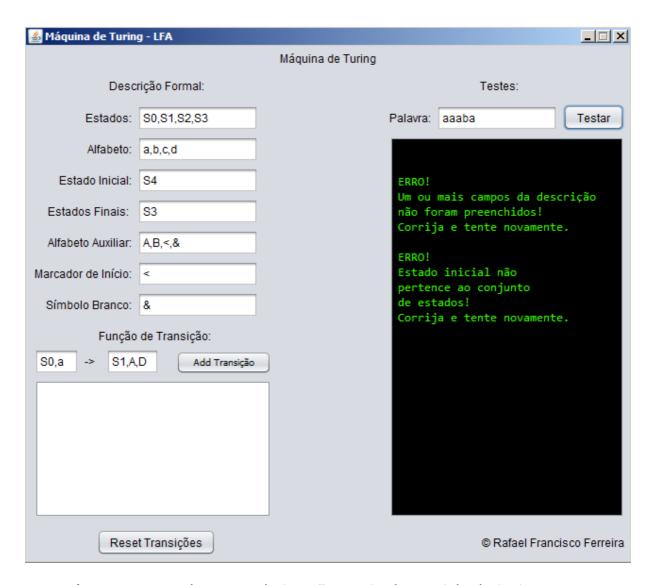
A máquina foi implementada de forma intuitiva, contendo campos para cada um dos itens da descrição formal.



Para utilizá-la, basta preencher os campos e adicionar a função de transição, feito isso, é só digitar uma palavra e clicar no botão "Testar".

Caso haja um erro de digitação nos campos ou nas transições, um alerta poderá ser gerado informando o que o causou.

Se o usuário cometer um equívoco e adicionar uma transição com algum parâmetro errado, basta clicar em "Reset Transições".



As mensagens de erro e alertas são mostrados na tela de testes sempre que o usuário tenta desempenhar uma ação e há algo de errado.

Se a máquina encontrar um erro durante o processamento da palavra, como um estado que não tem transição com a letra lida da fita, ela informará o erro, dizendo em qual estado parou e informar que a palavra não pertence à linguagem descrita.

Os campos são *case-sensitive*, ou seja, diferenciam letras maiúsculas de minúsculas. Deve-se estar atento ao adicionar alfabetos e nomear estados.

Para um melhor funcionamento da máquina, utilize sempre a vírgula para separar os nomes e símbolos dos alfabetos e conjuntos, sem espaços (!).

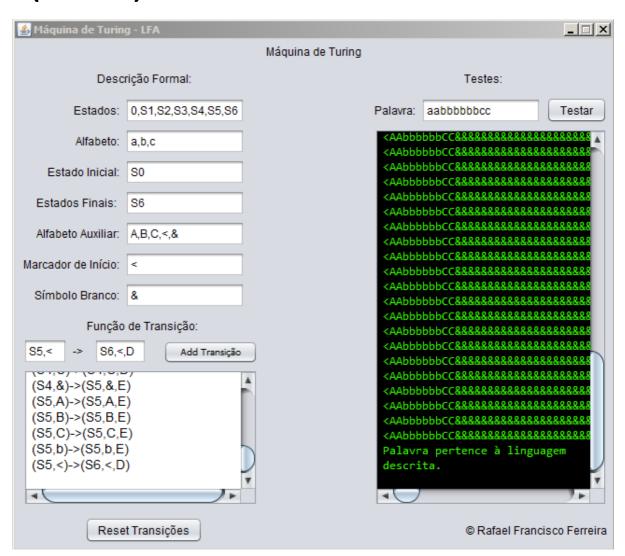
Ao abrir o programa, exemplos estarão pré-definidos nos campos de digitação. Caso queira mudar os símbolos, estes são escolha do usuário, mas lembre-se de manter a mesma formatação dos exemplos. Se algum símbolo digitado violar alguma regra ou condição, a máquina informará o erro.

4. Exemplos

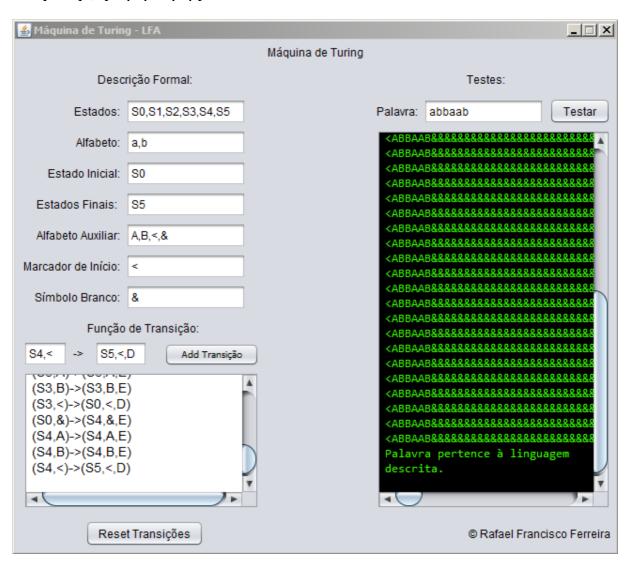
Abaixo estão alguns exemplos de utilização com linguagens aplicadas à máquina implementada.

Reconhecedoras:

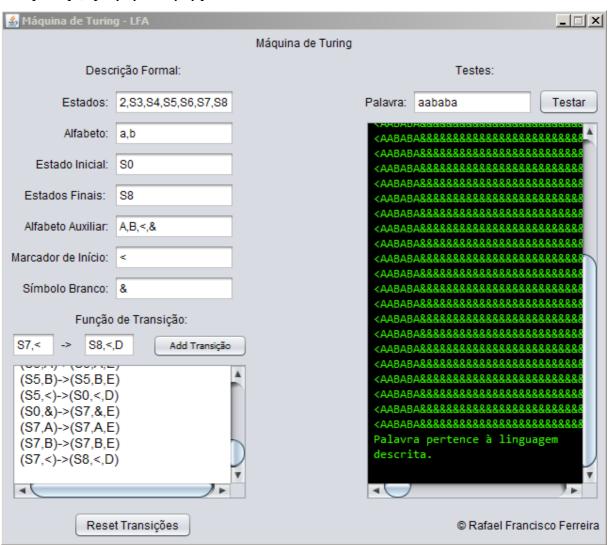
$L = \{a^nb^mc^n/n>=0\}$



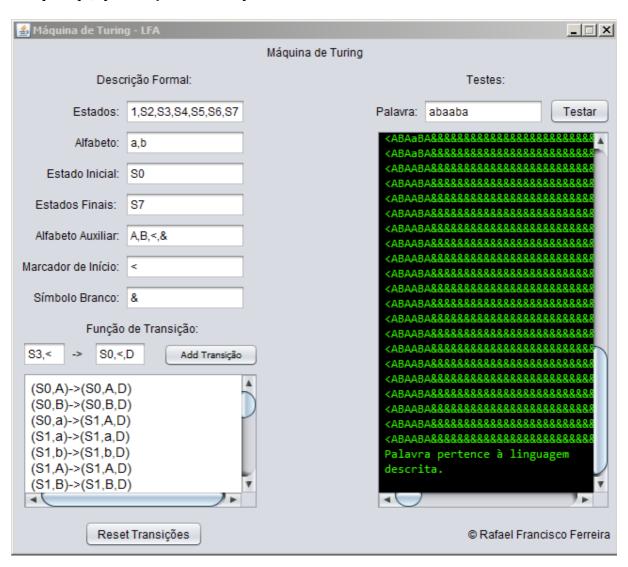
$L = \{w \in \{a,b\}^*/ |w|_a = |w|_b\}$



$L = \{w \in \{a,b\}^* / |w|_a = 2^* |w|_b\}$



L = $\{w \in \{a,b\}^*/ w \in palindromo\}$

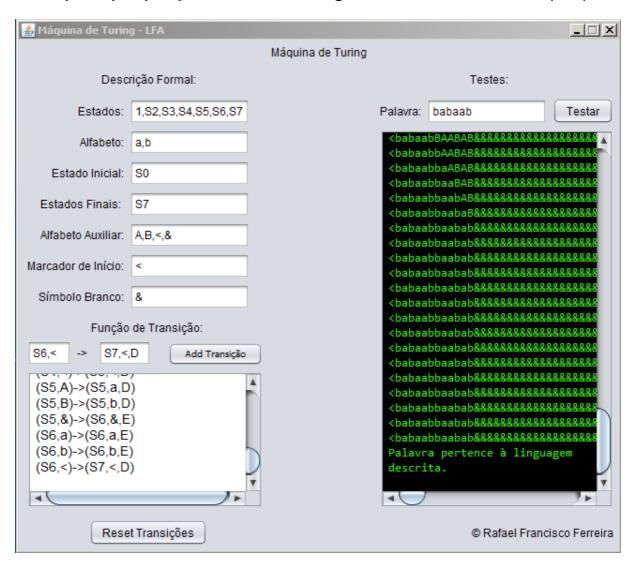


Transdutoras

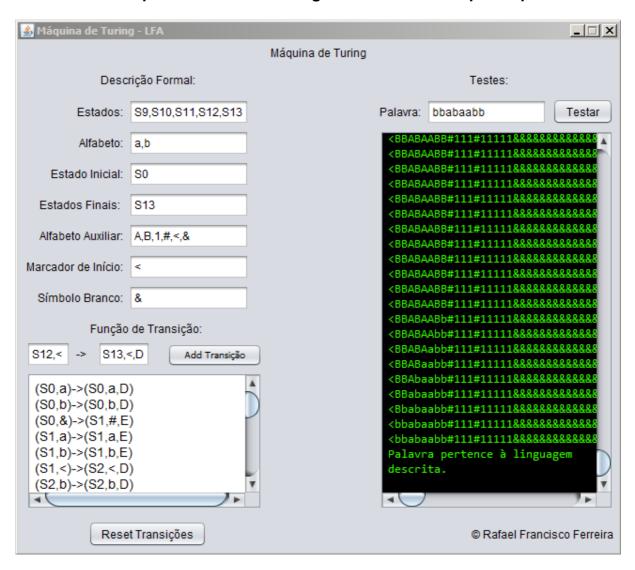
L = Recebe um número binário e o incrementa em uma unidade:

🚣 Máquina de Turin	g - LFA		_ IX
		Máquina de Turing	
Desc	rição Formal:		Testes:
Estados:	S0,S1,S2,S3,S4,S5	Palavra:	1011 Testar
Alfabeto:	0,1	<10118 <10118	222222222222222222222222222222222222222
Estado Inicial:	S0	<10118 <10118 <10118	
Estados Finais:	S3	<10118 <10118 <10118	88888888888888888888888888888888888
Alfabeto Auxiliar:	<,&	<1010 <10108 <10008	82222222222222222222222222222222222222
Marcador de Início:	<	<11008 <11008	3.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2.2
Símbolo Branco:	&		kaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
Função	de Transição:	descr	
S2,< >> S3,<			
(\$4,0)->(\$5,1,D (\$5,0)->(\$5,0,D (\$5,&)->(\$2,0,E (\$2,0)->(\$2,0,E (\$2,1)->(\$2,1,E (\$2,<)->(\$3,<,D			
4) · \	■) .
Rese	t Transições		© Rafael Francisco Ferreira

L = Duplica qualquer palavra com a e b seguida dela mesma invertida (ww')



L = Recebe uma palavra w com a e b e gera como saída w#qtdea#qtdeb



Exemplos de Tratamento de Erro

Caso o usuário esqueça um campo em branco:

♣ Máquina de Turino	a 154	sampo om bra		_ X
E Priaquilla de Turilli	y - LFA	Máquina de Turing		
		maqama ao ranng		
Descr	rição Formal:			Testes:
Estados:	S0,S1,S2,S3	F	Palavra:	aaaba Testar
Alfabeto:				
Estado Inicial:	80		ERRO!	mais campos da descrição
Estados Finais:	S3		não fo	ram preenchidos! a e tente novamente.
Alfabeto Auxiliar:	A,B,<,&		2025	
Marcador de Início:	<			
Símbolo Branco:	&			
Função	de Transição:			
S3,a -> S1,A,	D Add Transição			
Rese	t Transições			© Rafael Francisco Ferreira

Caso o estado inicial não pertença ao conjunto de estados:

🚣 Máquina de Turin	g - LFA				_
		Máquina de Turing			
Desc	rição Formal:			Testes:	
Estados:	00.04.00.03		Palavra:	aaaba	Testar
Estados.	S0,S1,S2,S3		Palavia.	aaaba	Testar
Alfabeto:	a,b				
Estado Inicial:	S4		ERRO!		
Fatadas Finsis:	62			inicial não	
Estados Finais:	S3		de est	ce ao conjunto ados!	
Alfabeto Auxiliar:	A,B,<,&		Corrij	a e tente novamente.	
Marcador de Início:	<				
Símbolo Branco:	&				
Função	de Transição:				
S0,a -> S1,A	,D Add Transição				
(S0,a)->(S1,A,D))				
Poso	et Transições			© Rafael Franciso	o Eorraira
Rese	t Hallsições			⊚ Raiaei Frailust	o renena

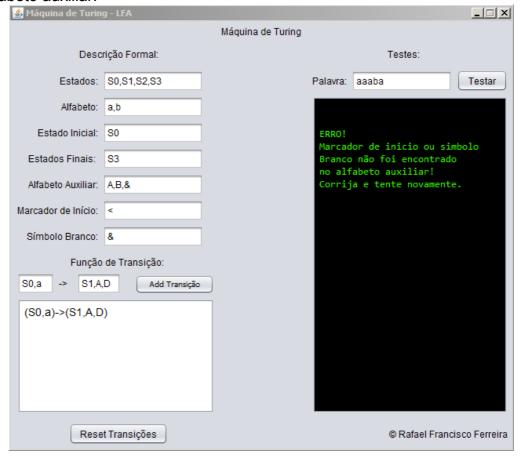
🚣 Máquina de Turing - LF/ Máquina de Turing Descrição Formal: Testes: Estados: S0,S1,S2,S3 Palavra: aaaba Testar Alfabeto: a,b Estado Inicial: S0 ERRO! Um ou mais estados Estados Finais: S4 finais não pertencem ao conjunto de estados! Corrija e tente novamente. Alfabeto Auxiliar: A.B.<.& Marcador de Início: < Símbolo Branco: & Função de Transição: S0,a -> S1,A,D Add Transição (S0,a)->(S1,A,D)

Caso um ou mais estados finais não pertençam ao conjunto de estados:

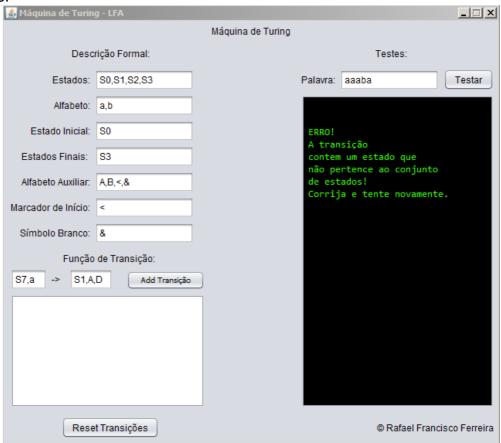
Caso o usuário esqueça de adicionar o marcador de inicio ou simbolo branco ao alfabeto auxiliar:

© Rafael Francisco Ferreira

Reset Transições



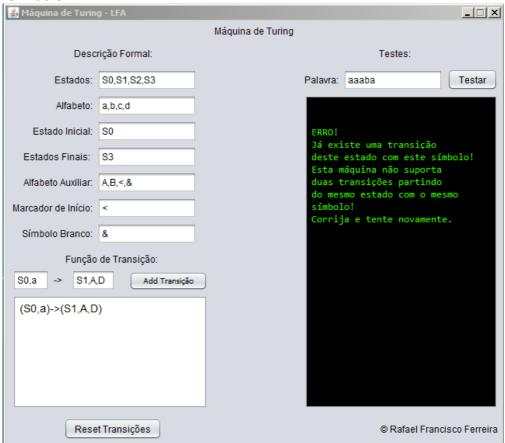
Caso a transição contenha um estado que não pertence ao conjunto de estados:



Caso a transição contenha um símbolo que não pertence aos alfabetos:

🚣 Máquina de Turin	g - LFA		_
		Máquina de Turing	
Desc	rição Formal:		Testes:
Estados:	S0,S1,S2,S3		Palavra: aaaba Testar
Alfabeto:	a,b,c,d		
Estado Inicial:	SO		ERRO! A transição contem
Estados Finais:	S3		um simbolo que não pertence a nenhum dos alfabetos!
Alfabeto Auxiliar:	A,B,<,&		Corrija e tente novamente.
Marcador de Início:	<		
Símbolo Branco:	&		
Função	de Transição:		
S0,X -> S1,A,	D Add Transição)	
Rese	t Transições		© Rafael Francisco Ferreira

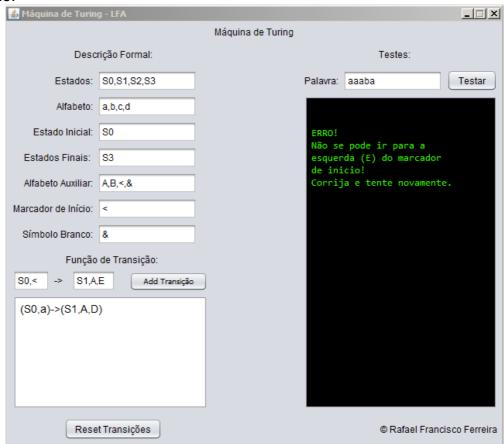
Caso o usuário tente criar duas transições partindo do mesmo estado com o mesmo símbolo:



Se o usuário tentar criar uma transição que alteraria o marcador de início:

🚣 Máquina de Turin	g - LFA			_ X
		Máquina de Turing		
Desc	rição Formal:		Testes:	
Estados:	S0,S1,S2,S3		Palavra: aaaba	Testar
Alfabeto:	a,b,c,d			
Estado Inicial:	S0		ERRO! Marcador de incio não	
Estados Finais:	S3		pode ser alterado na fita!	
Alfabeto Auxiliar:	A,B,<,&		corrija e tente novamente.	
Marcador de Início:	<			
Símbolo Branco:	&			
Função	de Transição:			
S0,< >> S1,A,	D Add Transição			
(S0,a)->(S1,A,D))			
Rese	t Transições		© Rafael Francisco	o Ferreira
	Estados: Alfabeto: Estado Inicial: Estados Finais: Alfabeto Auxiliar: Marcador de Início: Símbolo Branco: Função S0,< -> S1,A (S0,a)->(S1,A,D	Alfabeto: a,b,c,d Estado Inicial: S0 Estados Finais: S3 Alfabeto Auxiliar: A,B,<,& Marcador de Início: < Símbolo Branco: & Função de Transição:	Descrição Formal: Estados: \$0,\$1,\$2,\$3 Alfabeto: a,b,c,d Estado Inicial: \$0 Estados Finais: \$3 Alfabeto Auxiliar: A,B,<,& Marcador de Início: < Símbolo Branco: & Função de Transição: \$0,< -> \$1,A,D Add Transição (\$0,a)->(\$1,A,D)	Descrição Formal: Estados: S0,S1,S2,S3 Alfabeto: a,b,c,d Estado Inicial: S0 Estados Finais: S3 Alfabeto Auxiliar: A,B,<,& Marcador de Início: < Símbolo Branco: & Função de Transição: S0,< -> S1,A,D Add Transição (S0,a)->(S1,A,D)

Se o usuário tentar criar uma transição que iria para a esquerda do marcador de início:



Caso o símbolo de direção esteja fora dos padrões: (Estabelecidos como *D para direita e E para esquerda.*)

💪 Máquina de Turin	g - LFA		_ X	ĺ
		Máquina de Turing		
Desc	rição Formal:		Testes:	
Estados:	S0,S1,S2,S3		Palavra: aaaba Testar	
Alfabeto:	a,b,c,d			
Estado Inicial:	S0		ERRO!	
Estados Finais:	S3		Utilize apenas D para direita ou E para esquerda! Corrija e tente novamente.	
Alfabeto Auxiliar:	A,B,<,&		corriga e cence novamence.	
Marcador de Início:	<			
Símbolo Branco:	&			
Função	de Transição:			
S0,< -> S1,A	,d Add Transição			
(S0,a)->(S1,A,D))			
Rese	et Transições		© Rafael Francisco Ferreira	

Caso a máquina encontre um estado de erro, ela informa em qual estado foi:

🙆 Máquina de Turin	g - LFA			×
		Máquina de Turing		
Desc	rição Formal:		Testes:	
Estados:	S0,S1,S2,S3		Palavra: aaaba Testar	
Alfabeto:	a,b		<aaaba&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&< th=""><th>8,5</th></aaaba&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&<>	8,5
Estado Inicial:	S0		<pre><aaaba&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&< th=""><th></th></aaaba&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&<></pre>	
Estados Finais:	S1		Palavra não pertence à linguagem descrita.	ı
Alfabeto Auxiliar:	A,B,<,&			ı
Marcador de Início:	<			ı
Símbolo Branco:	&			ı
Função	de Transição:			ı
S0,a -> S1,A	D Add Transição			ı
(S0,a)->(S1,A,D))			ı
				ı
				ı
				ı
			4	2
Rese	rt Transições		© Rafael Francisco Ferre	ira

5. Decisões de Projeto

A primeira decisão importante foi a da linguagem de programação. A linguagem escolhida foi JAVA, levando em consideração, principalmente, que a mesma oferece uma grande variedade de opções para se trabalhar de forma prática com estruturas de dados, e que oferece a orientação a objetos.

As estruturas de dados utilizadas foram, em sua maioria, vetores, que armazenam informações como: alfabeto, alfabeto auxiliar, nomes dos estados, estados finais, palavra, partes das transições e etc. As outras estruturas utilizadas foram as Listas, que armazenam dados contidos em forma de objetos, tais como as Transições e os Estados, e também auxiliaram no armazenamento de Strings para tratamentos de erros e para consultas condicionais.

Os objetos foram criados para facilitar a implementação e organizar melhor o código. São objetos: MT, Fita, Transições, Estados. Cada objeto armazena suas respectivas informações privadas e públicas em estruturas de dados e variáveis locais. O objeto MT comanda todo o funcionamento da Máquina de Turing, sendo instanciado no momento que o usuário clica no botão "Testar".

A interface com o usuário foi desenvolvida utilizando formulário java (JForm), visando facilitar o uso e também facilitar a captação dos dados.

Todos os erros foram tratados antes de se criar a máquina, exceto aqueles que a própria máquina deve informar, para evitar possíveis travamentos da máquina por erros de digitação ou campos mal preenchidos pelo usuário.

6. Conclusão

Este relatório apresentou uma implementação de uma Máquina de Turing Reconhecedora e Transdutora.

Como se nota nos exemplos, a máquina proposta foi testada com as mais diversas linguagens e retornou os resultados corretos em todos os testes feitos.

Além dos exemplos apresentados, duas listas de linguagens foram testadas mais de uma vez na máquina durante o desenvolvimento, e a mesma não falhou em nenhum dos testes.

Conclui-se, então, que a máquina implementada é eficaz no reconhecimento e transformação das palavras nela inseridas e testadas.

7. Referências

Vieira, Newton José.; (2006) "Introdução aos Fundamentos da Computação", Pioneira Thomson Learning.