

EDITOR DE AUTOMATAS FINITOS

FELIPE S. QUADROS ANA M. PRICE

Universidade Federal do Rio Grande do Sul

Porto Alegre - Brasil

R E S U M O

Este artigo descreve um editor gráfico de diagramas de transição desenvolvido com o objetivo de motivar alunos ao estudo de autômatos finitos e auxiliá-los na solução de exercícios sobre reconhecedores de linguagens regulares. O editor é de uso extremamente fácil pois sua interface com o usuário compreende, praticamente, uma única tela, na qual estão indicados os modos de operação e os comandos disponíveis ao sistema.

1. INTRODUÇÃO

O editor de automatos faz parte do Laboratório de Compiladores que está sendo desenvolvido para dar suporte às disciplinas relativas ao ensino de compiladores, ministradas em ambos os cursos de Bacharelado e Pós-Graduação em Ciência da Computação da UFRGS. O Laboratório compõe-se, basicamente, de dois módulos:

I) um conjunto de programas que simulam máquinas de Turing e autômatos, cuja finalidade é motivar e auxiliar alunos na solução de exercícios sobre reconhecedores de linguagens tipo 0, 1, 2 e 3 [Hopcroft 69]. Acredita-se que a disponibilidade de simuladores interativos, fáceis de usar e com visual atraente,

servirão de estímulo ao estudo do tema e à solução de exercícios propostos; e

II) um conjunto de subrotinas que implementam algoritmos pertinentes ao processo de compilação, como por exemplo, algoritmos de simplificação de gramáticas, geradores de tabelas LL e LR, analisadores sintáticos, etc. [Aho 77]. Este módulo visa dar suporte aos projetos de compiladores a serem desenvolvidos pelos estudantes. Devido à exiguidade de tempo para implementação e à complexidade de construção de um compilador, a idéia é que os estudantes desenvolvam alguns módulos do compilador em projeto, integrando-os com módulos disponíveis no Laboratório.

O editor gráfico aqui descrito é um programa interativo, extremamente fácil de ser operado, cuja interface compreende uma única tela e uma dezena de comandos imediatos.

A construção de um autômato finito é efetuada através de operações simples, como por exemplo:

- para criar um estado do autômato, basta movimentar o cursor para a posição desejada na tela e apertar a tecla CR;
- para definir um arco entre dois nodos, devem ser pressionadas as teclas de movimentação do cursor, conforme o caminho desejado, e o arco será traçado.

O capítulo a seguir apresenta fundamentos sobre autômatos finitos. A interface do usuário com o editor é descrita no Capítulo 3. O capítulo seguinte explica como usar o editor para construir autômatos, e o Capítulo 5, apresenta os comandos disponíveis ao usuário. Considerações sobre a atual implementação constituem o Capítulo 6. Conclusões são apresentadas no Capítulo 7.

Segundo [Hopcroft 69], um autômato finito é definido como um sistema $M = (K, \Sigma, d, q_0, F)$ onde

K representa um conjunto finito de estados,

Σ é um alfabeto finito de entrada,

d é uma função de transição, $d: (K \times \Sigma) \rightarrow K$,

q_0 é o estado inicial, e

F é o conjunto de estados finais.

Um autômato finito pode, também, ser representado através de um diagrama de transição em que os nodos representam estados e os arcos, a função de transição. O estado inicial é indicado, explicitamente, por uma seta inclinada, e os estados finais são designados por circunferências concêntricas. A Figura 1 ilustra um autômato finito que reconhece o conjunto de sentenças formadas por um número par de 0's (zeros) e 1's (uns) [Hopcroft 69].

$M = (K, \Sigma, d, q_0, F)$

$\Sigma = \{0, 1\}$

$K = \{q_0, q_1, q_2, q_3\}$

$F = \{q_0\}$

$d(q_0, 0) = q_1$

$d(q_0, 1) = q_3$

$d(q_1, 0) = q_0$

$d(q_1, 1) = q_2$

$d(q_2, 0) = q_3$

$d(q_2, 1) = q_1$

$d(q_3, 0) = q_2$

$d(q_3, 1) = q_0$

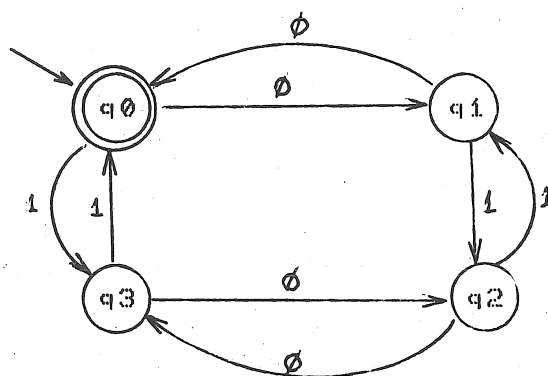


Figura 1

O editor gráfico trabalha com uma representação de autômato praticamente idêntica à descrita acima, com exceção de que os

estados são denotados por números e o estado inicial deve¹¹, obrigatoriamente, ser denotado por 0 (zero). Esta última suposição não impõe quaisquer restrições sobre a representação adotada, mas por outro lado, simplificou a implementação do editor. O autômato exemplo definido acima é mostrado, através da tela do editor gráfico, na Figura 2. (A indicação de estados finais ainda não foi implementada).

3. A TELA DE INTERFACE

A tela de interface com o usuário compõe-se de três partes distintas:

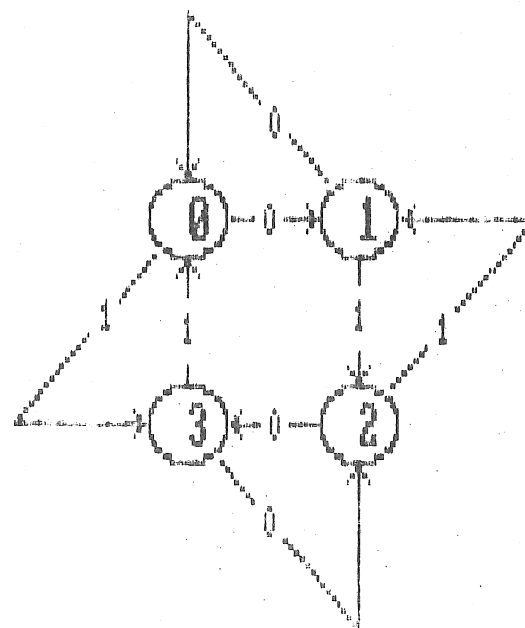
I) a área superior, que ocupa a maior parte da tela, está reservada para o autômato finito em edição. Internamente, esta área é representada como um reticulado de cinco linhas por dez colunas. Os vértices do reticulado indicam posições possíveis para nodos do diagrama, e as arestas e diagonais entre vértices podem conter arcos ou segmentos de arcos do autômato;

II) uma linha para a FITA DE LEITURA; e

III) a área inferior, usada para indicar o modo de operação do editor (EDITANDO, CAMINHANDO, APAGANDO), e mostrar os comandos disponíveis ao usuário, os quais foram implementados através das teclas de função F1 a F10.

4. CONSTRUINDO UM AUTÔMATO

Os comandos do editor (descritos no Capítulo 5) executam funções, tais como, gravar e remover autômato de disco, carregar fita de leitura e listar diretório, etc. A construção efetiva de



TELA DE LETURA

Modo: Caminhando
Editando
Apagando

F1 - Carrega
F2 - Salva
F3 - Diretorio
F4 - Executa

F5 - Carrega Fita
F6 - Limpa Fita
F7 - Remove
F8 - Imprime

F9 - Apaga
F10 - Fim
ESC - Ajuda

FIGURA 2

um autômato é realizada com o editor funcionando sob um dos modos de operação EDITANDO, CAMINHANDO ou APAGANDO. Quando operando sob um desses modos, as principais teclas para o editor são: CR (Carriage Return), CTRL F (Control F) e as setas de movimentação do cursor (inclusive as setas oblíquas), conforme pode-se perceber pela tabela abaixo. A passagem de um modo para outro é efetuada pela barra de espaço.

MODO DE OPERAÇÃO	CR	SETAS	CTRL F
EDITANDO	CRIA ESTADO OU TROCA NOME DE ESTADO	CRIA ARCO OU TROCA NOME DE ARCO	MARCA ESTADO FINAL
CAMINHANDO		MOVE CURSOR	
APAGANDO	REMOVE ESTADO	APAGA ARCO	DESMARCA ESTADO FINAL

MODO CAMINHANDO:

As teclas CR e CTRL F não produzem efeito algum sobre o autômato em edição, enquanto que, as SETAS deslocam o cursor de um vértice a outro do reticulado. Cabe lembrar que o reticulado é transparente ao usuário.

MODO EDITANDO:

A tecla CR causa a criação de um estado, caso o cursor não esteja posicionado sobre estado algum. Caso contrário, CR faz com que o editor aceite um novo nome para o estado sob o cursor.

As setas de movimentação do cursor geram arcos na direção e

sentido correspondentes, caso o cursor esteja sobre um estado do diagrama e não exista ainda qualquer arco na direção e sentido da SETA pressionada. Se existir tal arco, o editor fica apto a trocar o símbolo indicado no arco.

CTRL F marca o estado sob o cursor como um estado final do autômato.

MODO APAGANDO:

CR causa a extinção do estado sob o cursor se este nodo estiver completamente isolado do diagrama. Isto é, para remover-se um estado do autômato é necessário, em primeiro lugar, apagar todos os arcos que chegam ou saem deste estado. Caso contrário, não ocorre qualquer modificação sobre o autômato.

CTRL F desmarca o estado sob o cursor como estado final do autômato.

As SETAS de movimentação do cursor apagam um arco do diagrama quando o cursor estiver posicionado sobre um estado e existir um caminho na direção e sentido da tecla pressionada. Caso contrário, não acontece edição alguma sobre o autômato.

5. OS COMANDOS DO EDITOR

Os comandos implementam as funções de ler, gravar, remover autômatos de disco, listar diretório, imprimir diagrama, etc., através das teclas de função (F1 a F10) do teclado. São os seguintes:

CARREGA (F1)

Busca em disco um autômato previamente definido e carrega na memória. O nome do autômato é requisitado ao usuário através de

uma janela posicionada à direita da tela. O autômato solicitado é mostrado na tela ficando disponível para execução, edição ou quaisquer outras operações disponíveis no editor.

SALVA (F2)

Armazena em disco o autômato definido na tela, sob o nome especificado pelo usuário, em janela indicada no lado esquerdo da tela (Figura 3).

DIRETÓRIO (F3)

Mostra, em uma janela central, a lista de nomes de autômatos até então catalogados.

EXECUTA (F4)

Inicia o reconhecimento da sentença carregada na FITA DE LEITURA. Cada símbolo da fita ao ser lido fica em vídeo reverso, bem como o estado corrente do autômato em execução. O estado inicial é, por convenção, o estado 0. Os estados finais são indicados por um círculo com duas barras verticais internas. O reconhecimento ou não da sentença em análise é indicado através de mensagem na tela.

CARREGA FITA (F5)

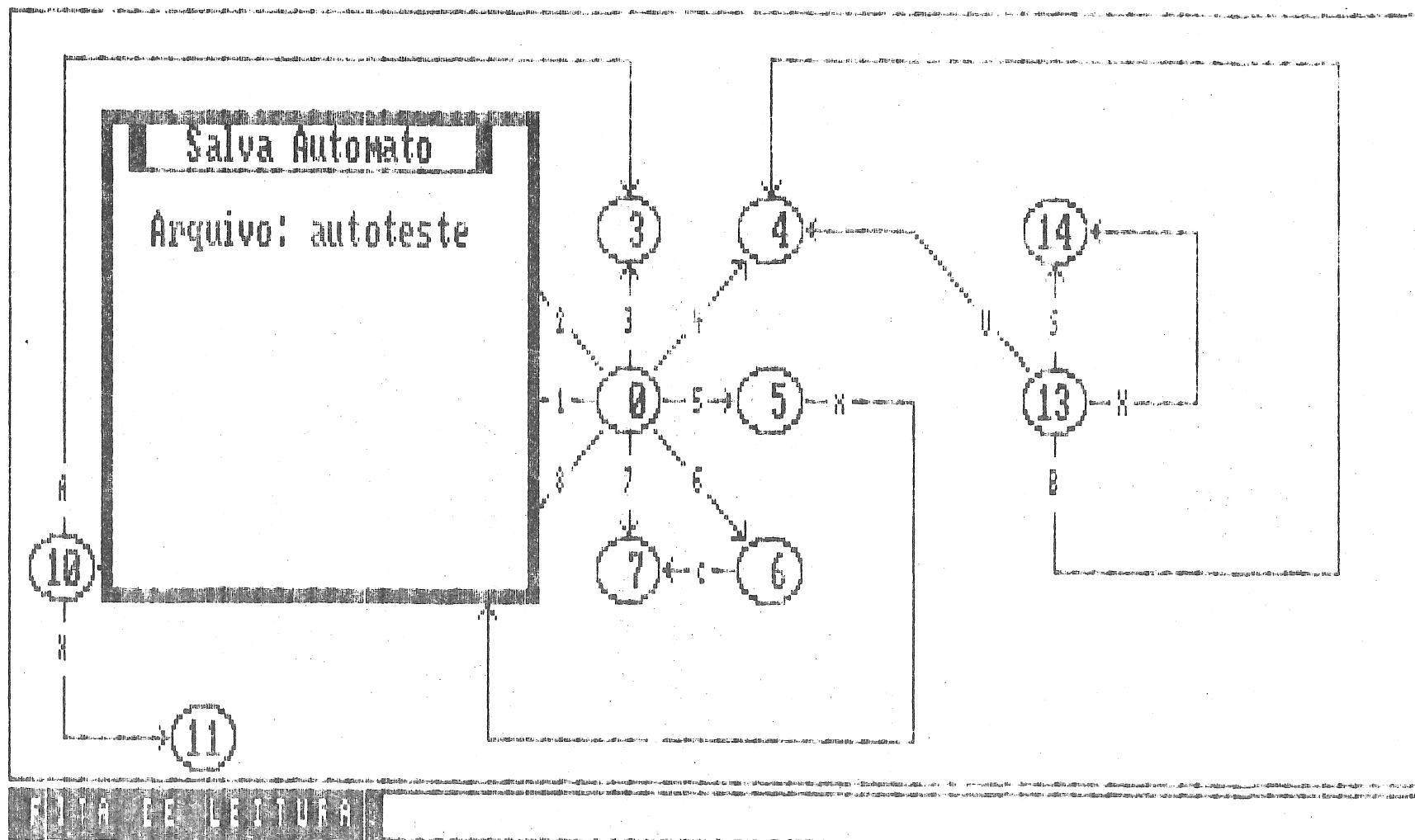
O editor fica apto a receber os símbolos da sentença a ser analisada.

LIMPA FITA (F6)

Preenche a FITA DE LEITURA com espaços.

REMOVE (F7)

Remove do disco o autômato nomeado pelo usuário, através de



Modo: Caminhando Editando Apagando	F1 - Carrega F2 - Salva F3 - Diretorio F4 - Executa	F5 - Carrega Fita F6 - Limpa Fita F7 - Remove F8 - Imprime	F9 - Apaga F10 - Fim ESC - Ajuda

FIGURA 3

uma janela central (superior)..

IMPRIME (F8)

Produz, na impressora, uma cópia do autômato mostrado na tela..

APAGA (F9)

Apaga o autômato mostrado na tela..

FIM (F10)

Encerra a sessão..

AJUDA (ESC)

Mostra na tela o manual do usuário..

As janelas de mensagens são distribuídas na tela em posições distintas para que o usuário associe a posição da janela com a função solicitada..

6. CONSIDERAÇÕES SOBRE A IMPLEMENTAÇÃO

O editor foi desenvolvido praticamente em Turbo Pascal, em microcomputadores do tipo PC, contendo, porém, algumas rotinas de manipulação gráfica em Assembler para aumentar o desempenho do sistema. O código fonte contém atualmente 1900 linhas..

As funções CARREGA FITA, EXECUTA e CTRL F (marca e desmarca estado final) ainda não estão disponíveis na atual implementação.. Por outro lado, a edição de autômatos está completamente operacional.. A implementação atual ainda não permite a escrita de

7. CONCLUSÕES

Acredita-se que a disponibilidade de ferramentas interativas de apoio ao ensino estimulam o estudo do tema associado e auxiliam a solução de problemas propostos em classe. O editor gráfico de autômatos finitos descrito neste relatório foi desenvolvido com esse objetivo.

O editor implementado é um programa interativo e fácil de ser operado devido à simplicidade de sua interface com o usuário: uma única tela, onde estão indicados os modos de operação e os comandos disponíveis, os últimos implementados através de teclas de função.

A construção de um autômato é realizada basicamente com o auxílio das teclas CR, CTRL F e as setas de movimentação do cursor. Imediatamente após a criação de um estado ou arco de autômato o editor exige a identificação do elemento criado, mantendo pois o diagrama sempre consistente.

O editor oferece, também, facilidades para gravação, impressão e catalogação de autômatos, e futuramente, a possibilidade de consulta ao manual do usuário.

Pretende-se introduzir esta ferramenta como suporte ao ensino de reconhecedores de linguagens regulares nos cursos de graduação e pós-graduação no próximo semestre letivo.

AGRADECIMENTOS

Os autores agradecem os incentivos recebidos do Conselho

Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

REFERÊNCIAS BIBLIOGRÁFICAS

[Hopcroft 69]

J. Hopcroft & J. Ullman, Formal Languages and Their Relation to Automata, Addison-Wesley, 1969.

[Aho 77]

A. Aho & J. Ullman, Principles of Compiler Design, Addison-Wesley, 1977.

S. Carvalho, Turbo Pascal, Version 3.0 Reference Manual, Borland International, 1985.