

Relatório de Aspectos Teóricos da Computação

Rafael da Silva Macêdo¹

¹ *Departamento de Ciências Exatas, Universidade Estadual de Feira de Santana*

3011rafael@gmail.com

28 de Outubro de 2014

1 Introdução

O conceito de autômatos é utilizado em diversas aplicações no nosso dia-a-dia. Além de estar presente nos computadores, estes estão aplicados em máquinas de refrigerantes, portas e portões automáticos, elevadores, jogos eletrônicos, entre outros.

Este documento faz um detalhamento da implementação de um autômato determinístico de estados finitos que simula o funcionamento de um elevador para um prédio com três andares (térreo, primeiro andar e segundo andar). A linguagem de programação utilizada na implementação do autômato foi C++.

Este relatório está dividido nos seguintes tópicos: *Expressões Regulares*, que faz uma fundamentação teórica sobre expressões regulares e por fim exemplifica com um exemplo de uma expressão no Vim; *Metodologia*, a qual explana a metodologia utilizada no processo de desenvolvimento do trabalho; *Resultados e discussões*, a qual faz um apanhado geral da implementação do autômato; *Manual do Usuário*, a qual tira algumas dúvidas do usuário e explana as principais regras para o bom funcionamento do software; e *Conclusões*.

2 Expressões Regulares

Expressões regulares são utilizadas para identificar, substituir e fazer modificações em um conjunto de caracteres. Estas podem ser descritas utilizando uma linguagem. Expressões regulares são bastantes utilizadas, pois permitem que sejam feitas inúmeras modificações em um texto com uma eficácia muito grande. Com expressões regulares é possível, por exemplo, transformar todas as primeiras letras de um parágrafo em maiúsculas.

As expressões criadas foram feitas utilizando a linguagem suportada pelo Vim, o qual é um interpretador de expressões regulares. As expressões foram criadas para dar uma formatação melhor em um texto, a primeira adiciona um espaçamento (Tab) no início de cada parágrafo do texto e a segunda procura todos os pontos do arquivos e torna maiúscula a primeira letra após o ponto. As expressões são bastante simples e devem ser executadas no Vim, tanto no Linux quanto no Windows.

Nas imagens abaixo é possível visualizar as expressões criada com os devidos comandos para

adicionar um espaçamento ao início de cada parágrafo do texto e para tornar a letra após um ponto maiúscula.

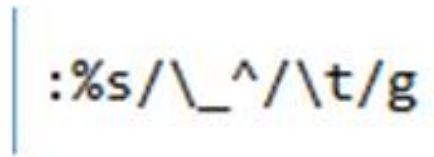
A regular expression pattern displayed on a light blue background. The pattern is `:%s/_^\t/g`. It starts with a colon, followed by a percent sign, then the letter 's', a backslash, an underscore, a caret, a backslash, the letter 't', a forward slash, and finally the letter 'g'.

Figura 1: Expressão Regular 1

A regular expression pattern displayed on a light blue background. The pattern is `:g/^\./s/^\./\u&/g|`. It starts with a colon, followed by the letter 'g', a forward slash, a caret, a period, a forward slash, the letter 's', a forward slash, a caret, a period, a forward slash, the letter 'u', an ampersand, a forward slash, the letter 'g', and ends with a vertical bar.

Figura 2: Expressão Regular 2

3 Metodologia

Para desenvolver o trabalho foi necessária a realização de uma pesquisa sobre as principais aplicações de autômatos no cotidiano das pessoas. Após esta fase determinou-se que seria implementado um autômato que simulasse o funcionamento de um elevador. Esta escolha foi feita por que esta aplicação é muito interessante e bastante utilizada no dia a dia das pessoas. A ilustração do autômato, a definição dos estados, transições e caracteres aceitos foram de suma importância para a concretização do trabalho realizado e realização de testes. Sem essas definições não seria possível a construção do código-fonte.

4 Resultados e Discussões

Neste trabalho foi feita a implementação de um autômato determinístico de estados finitos para simular o funcionamento de um elevador para um prédio com três andares. O autômato implementado possui seis estados, sendo um o inicial e três finais. O alfabeto do autômato é $\Sigma(0, 1, 2)$. Caso seja informado algum caractere que não pertence ao alfabeto o autômato não aceita e exibe mensagem de erro. Além dos seis estados principais do autômato, há um estado implícito que possui a funcionalidade de um sumidouro. O sumidouro é utilizado, se necessário, para limpar a cadeia de entrada e permitir que o usuário informe uma nova cadeia. O sumidouro é utilizado sempre que o usuário informa um caractere que não seja aceito pelo estado atual ou que não faça parte do alfabeto.

Determinou-se que o elevador só inicia seu funcionamento no andar térreo e quando sua porta estiver aberta. A partir do andar inicial o elevador pode chegar a qualquer um dos outros andares com a sua porta fechada, mas só poderá encerrar o percurso quando a porta for aberta. Para permanecer no mesmo andar e alterar apenas o estado da porta (aberta ou fechada) o autômato consome o caractere '0', para avançar um andar consome o caractere '1' e para

recuar um andar consome o caractere '2'.

Para a nomenclatura dos estados do autômato utilizou-se as seguintes siglas: TA (Andar Térreo com a Porta Aberta), TF (Andar Térreo com a Porta Fechada), 1A (Primeiro Andar com a Porta Aberta), 1F (Primeiro Andar com a Porta Fechada), 2A (Segundo Andar com a Porta Aberta) e 2F (Segundo Andar com a Porta Fechada). A ilustração do autômato implementado com os seus estados e transições pode ser visualizada na Figura 3.

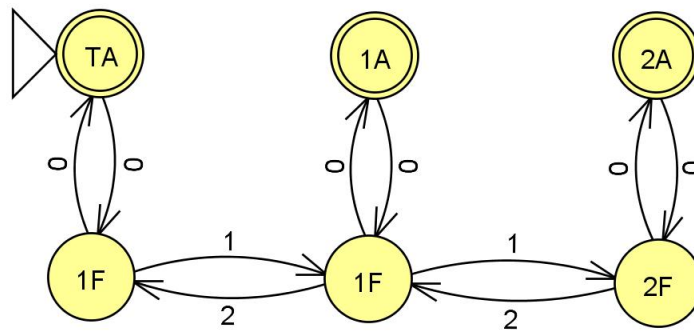
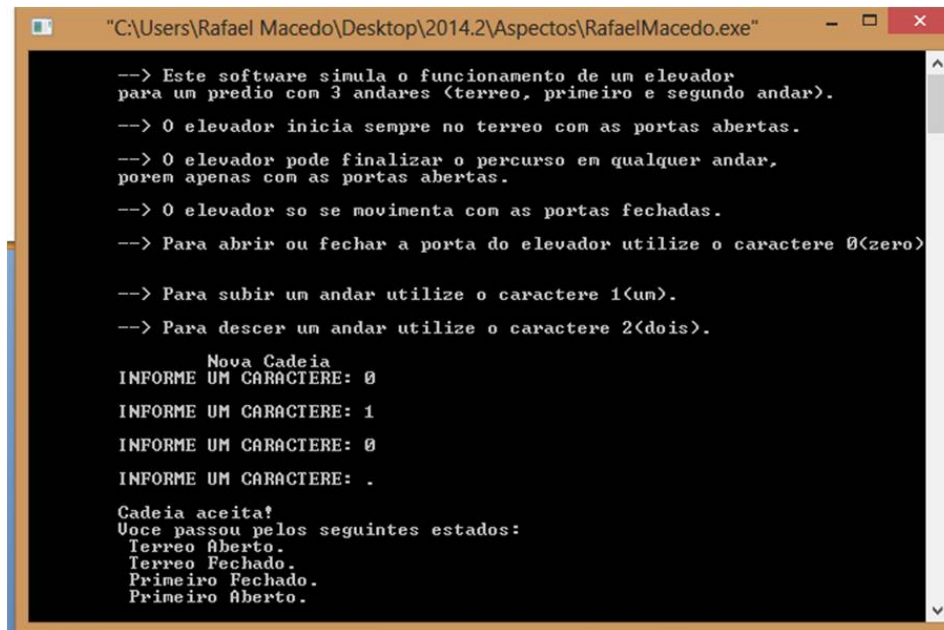


Figura 3: Automato Elevador

O software possui apenas uma tela, nesta deve ser informada a cadeia de entrada e na mesma são mostradas as saídas. O software implementado possui dois tipos de saída: uma quando todos os caracteres da cadeia são aceitos (exibida mensagem de sucesso e lista de estados) e quando é informada uma cadeia inválida (exibida mensagem de erro). A seguir, na Figura 4, há uma tela com um exemplo de cadeia aceita, e logo em seguida, na Figura 5 foi informada uma cadeia inválida por isso foi exibida mensagem de erro.



```
"C:\Users\Rafael Macedo\Desktop\2014.2\Aspectos\RafaelMacedo.exe"

--> Este software simula o funcionamento de um elevador
para um predio com 3 andares (terreo, primeiro e segundo andar).

--> O elevador inicia sempre no terreo com as portas abertas.

--> O elevador pode finalizar o percurso em qualquer andar,
porem apenas com as portas abertas.

--> O elevador so se movimenta com as portas fechadas.

--> Para abrir ou fechar a porta do elevador utilize o caractere 0<zero>

--> Para subir um andar utilize o caractere 1<um>.

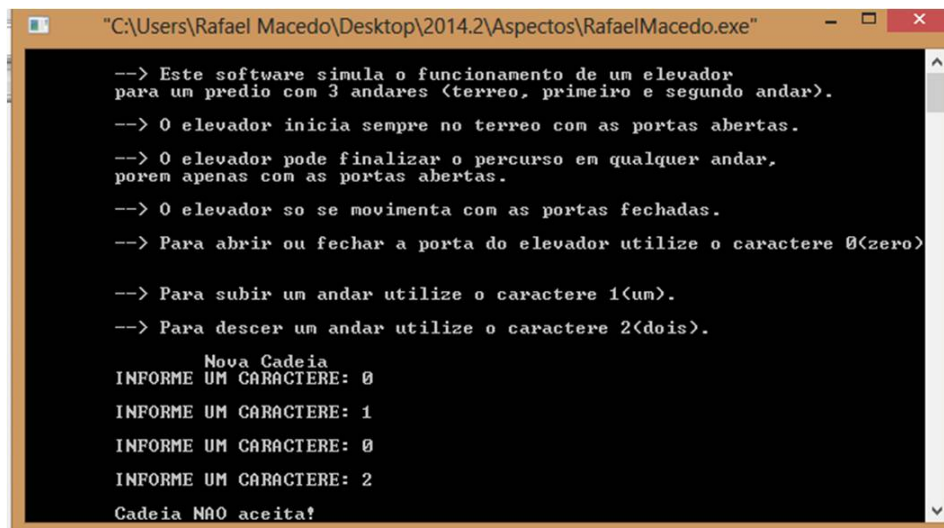
--> Para descer um andar utilize o caractere 2<dois>.

      Nova Cadeia
INFORME UM CARACTERE: 0
INFORME UM CARACTERE: 1
INFORME UM CARACTERE: 0
INFORME UM CARACTERE: .

Cadeia aceita!
Voce passou pelos seguintes estados:
Terreo Aberto.
Terreo Fechado.
Primeiro Fechado.
Primeiro Aberto.
```

Figura 4: Execução do autômato com cadeia aceita

Na figura acima foi informada uma cadeia para que o elevador se movimenta-se até o primeiro andar. Os caracteres iniciais e finais da cadeia servem apenas para fechar e abrir a porta do elevador respectivamente. Visto que a cadeia é aceita o software informou a mensagem: "Cadeia aceita" e logo em seguida a lista de todos os estados que o elevador se movimentou juntamente com a situação da porta (aberta ou fechada).



```
"C:\Users\Rafael Macedo\Desktop\2014.2\Aspectos\RafaelMacedo.exe"

--> Este software simula o funcionamento de um elevador
para um predio com 3 andares (terreo, primeiro e segundo andar).

--> O elevador inicia sempre no terreo com as portas abertas.

--> O elevador pode finalizar o percurso em qualquer andar,
porem apenas com as portas abertas.

--> O elevador so se movimenta com as portas fechadas.

--> Para abrir ou fechar a porta do elevador utilize o caractere 0<zero>

--> Para subir um andar utilize o caractere 1<um>.

--> Para descer um andar utilize o caractere 2<dois>.

      Nova Cadeia
INFORME UM CARACTERE: 0
INFORME UM CARACTERE: 1
INFORME UM CARACTERE: 0
INFORME UM CARACTERE: 2
Cadeia NAO aceita!
```

Figura 5: Execução do autômato com cadeia inválida

Na figura acima foi informada uma cadeia inválida. A cadeia informada não é aceita pelo autômato devido ao último caractere que foi '2'(dois), mas o estado atual era o 1A, o qual só aceita o caractere '0' zero. Como nesse caso, sempre que a cadeia não é válida é exibida a mensagem de erro: "Cadeia NAO aceita!".

5 Manual do Usuário

O software implementado que simula o funcionamento do autômato de um elevador é bastante simples. O usuário deve informar uma sequência de caracteres que pertençam ao alfabeto aceito pelo autômato $\Sigma(0, 1, 2)$. Se o usuário informar um caractere diferente destes não será aceito e será exibida a mensagem de erro: "Cadeia Não Aceita".

O estado inicial do autômato é o TA (Andar Térreo com a Porta Aberta). A partir deste estado é possível a navegação por todos os outros estados, se for obedecido a lógica do mesmo. Algumas regras:

1. O caractere inicial de toda cadeia deverá ser '0'. Isto garante que o elevador só se movimentará com as portas fechadas;
2. O caractere final de toda cadeia deverá ser '0'. Isto garante que o elevador chegará ao fim do percurso com suas portas abertas;
3. É permitido a passagem pelos estados (andares) quantas vezes forem necessárias;
4. Para avançar um andar utilize o caractere '1';
5. Para recuar um andar utilize o caractere '2'.

O software implementado permite que o usuário digite diversas cadeias, uma por vez e um caractere por vez. Para cada cadeia será exibida na tela uma mensagem informando se a cadeia é aceita pelo autômato.

A utilização do software é bastante simples, basta informar uma sequência de caracteres, um por vez pressionando a tecla enter após o caractere. Para informar o fim da cadeia deve ser inserido o caractere '.'(ponto). Após o pressionamento da tecla '.' será exibida uma mensagem de retorno informando se a cadeia é aceita. Caso a cadeia seja aceita será exibida uma lista com todos os estados transitados. Após a impressão da mensagem o usuário poderá informar outra sequência de caracteres independente da(s) anterior(es).

6 Conclusões

Ao final deste trabalho percebe-se que há infinitas aplicações que se adequam ao conceito de autômatos. Com isso percebe-se a importância desses elementos no cotidiano das pessoas. Escolheu-se a implementação do autômato simulador de um elevador devido à importância

desses instrumentos do dia-a-dia das pessoas e por seus funcionamentos serem bastantes conhecidos pelas pessoas.

Também ficou bastante explanado a importância das expressões regulares nos dias atuais, pois com a utilizações de alguns comandos é possível realizar tarefas, que demandam bastante tempo quando feitas manualmente, em um curto espaço de tempo.