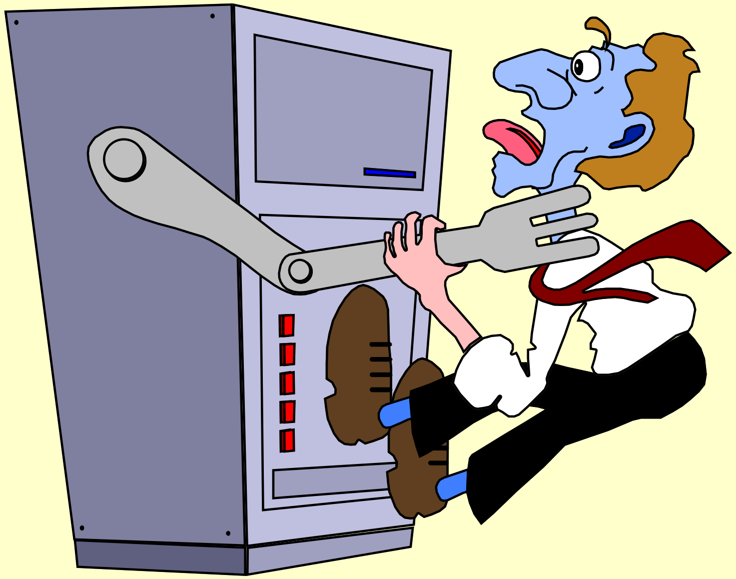
Máquinas de Turing



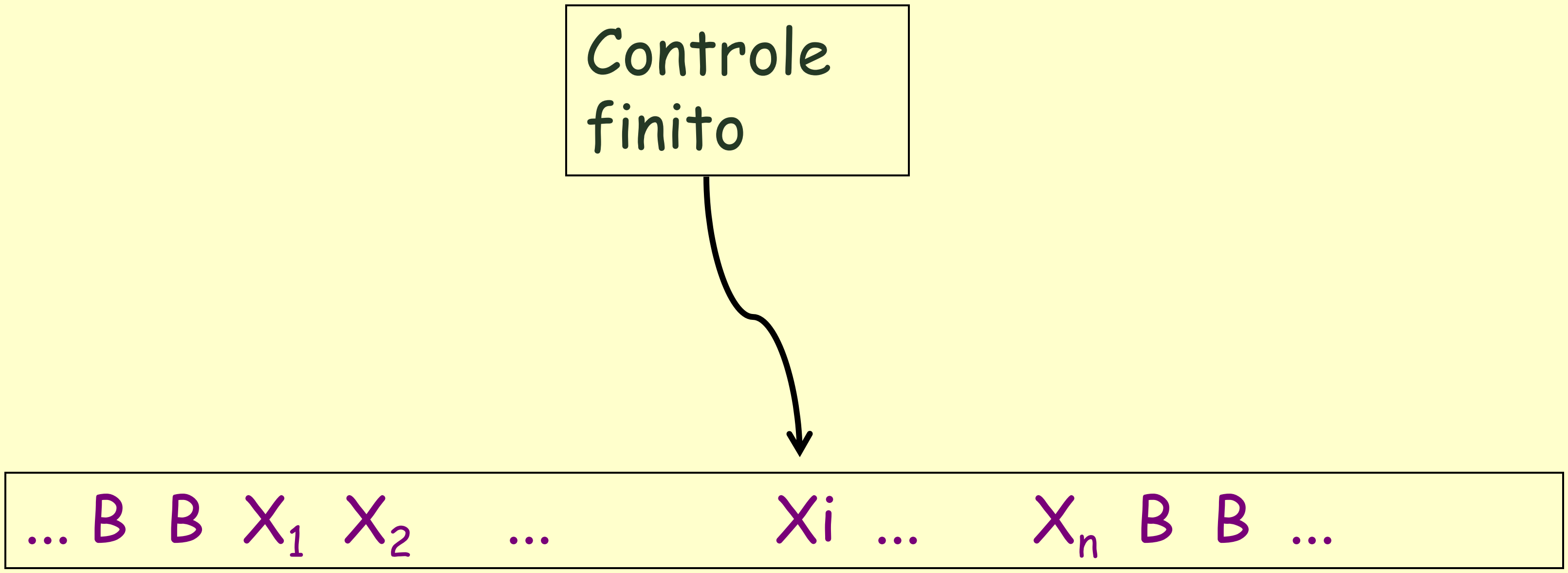
Máquinas de Turing podem fazer tudo o que um computador real faz.

Porém, mesmo uma Máquina de Turing não pode resolver certos problemas. Estes problemas estão além dos limites teóricos da computação

História

- Turing (1936): Máquinas de Turing como modelo de função computável.
- Tese de Church-Turing: qualquer modelo geral de computação permite calcular as mesmas funções (ou, tudo o que se pode computar coincide com as linguagens reconhecidas pelas Máquinas de Turing).

Máquina de Turing



Inicialmente, a entrada é colocada na fita. Todas as outras células (infinitamente à esquerda e à direita) têm um símbolo especial da fita, B (*branco*).
A cabeça da fita fica posicionada em uma das células. No início, a cabeça está posicionada na célula mais à esquerda que contém a entrada.

Um *movimento* da MT é uma função do estado do controle finito e do símbolo atual da fita. Em um movimento, a MT:

1. Mudará de estado (opcionalmente para o mesmo).
2. Gravará um símbolo de fita na célula atual, substituindo o existente (podendo ser o mesmo).
3. Movimentará (necessariamente) a cabeça da fita uma célula à esquerda ou à direita.

MT: notação formal

M = (Q, Σ, Γ, δ, qo, F)

Controle finito

Q=conj. finito de estados;
F = conj. estados finais (de aceitação)

Γ = alfabeto finito da fita

cabeça da fita

... B B X_1 X_2 Xi X_n B B ...

Σ = alfabeto finito de entrada

Máquina de Turing

Função de transição δ:

$\delta: Q \times \Gamma \rightarrow Q \times \Gamma \times \{L,R\}$

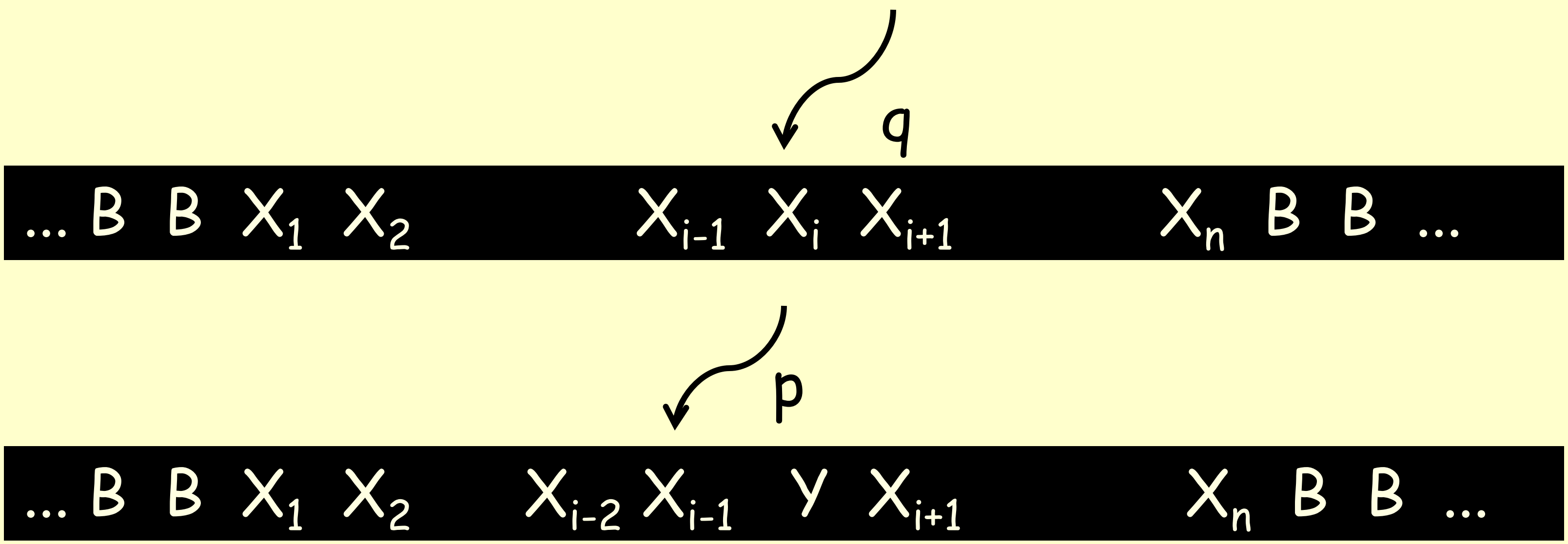
Ou seja, $\delta(q,X) = (p,Y,D)$ onde:

- p é o próximo estado em Q;
- Y é o símbolo que substituirá X na fita;
- D é uma direção (esquerda ou direita) em que a cabeça da fita irá se mover.

Descrições Instantâneas para MT

Suponha que $\delta(q,X_i) = (p,Y,L)$, ou seja, o movimento foi para a esquerda. Então:

$$X_1X_2....X_{i-1}qX_iX_{i+1}\ ...X_n \vdash_M X_1X_2....X_{i-2}pXi_{-1}YX_{i+1}...X_n$$



Duas exceções:

- Se i = 1, então M se move para o B à esquerda de X1. Nesse caso:

$$qX_1X_2 \ldots X_n \vdash pBYX_2 \ldots X_n$$

- Se i = n e Y = B, então o B gravado sobre X_n se junta ao sufixo de Bs e não aparece na próxima DI:

$$X_1X_2 \ldots X_{n-1}qX_n \vdash X_1X_2 \ldots X_{n-2}pX_{n-1}$$

Agora suponha que $\delta(q,X_i) = (p,Y,R)$, ou seja, o movimento foi para a direita. Então:

$$X_1X_2....X_{i-1}qX_iX_{i+1} ...X_n \underset{M}{\vdash} X_1X_2....X_{i-1}YpX_{i+1}...X_n$$

Duas exceções:

- Se i = n, então a (i+1)-ésima célula contém um B e ela não faz parte da DI anterior. Nesse caso:

  $$X_1X_2 ...X_{n-1}qX_n \vdash X_1X_2 ... X_{n-1}YpB$$

- Se i = 1 e Y = B, então o B gravado sobre X1 se junta ao prefixo de Bs e não aparece na próxima DI:

  $$qX_1X_2 ...X_n \vdash pX_2 ... X_n$$

Exemplo

- Vamos projetar uma MT para reconhecer

  $L = \{0^n1^n \mid n \geq 1\}$

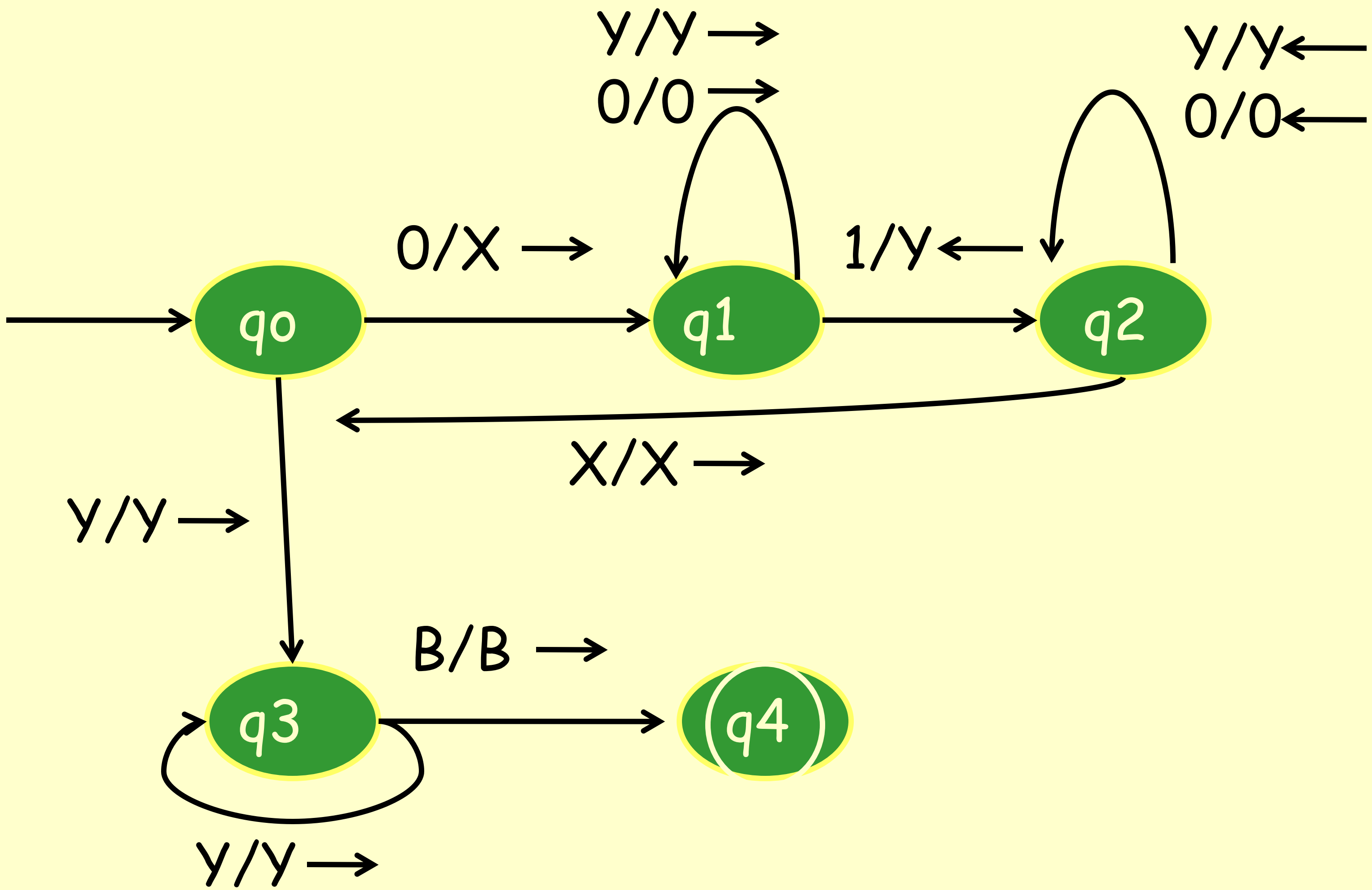
- Estratégia: a MT trocará um 0 por um X, e depois um 1 por um Y, até todos os 0s e 1s terem sido comparados.
- Em cada passo, da esq. para dir., ela troca um 0 por X e vai para a direita, ignorando 0s e Ys até encontrar 1. Troca esse 1 por Y e se move para a esquerda, ignorando Ys e 0s, até encontrar um X. Procura um 0 a direita e troca por X, repetindo o processo.
- Se a entrada não estiver em 0^n1^n eventualmente a MT não vai ter um movimento previsto e vai parar sem aceitar.
- Se, por outro lado, na busca por mais um 0, ela só encontrar Xs e Ys, então ela descobre que deve aceitar a entrada.

$M = (\{q0,q1,q2,q3,q4\}, \{0,1\}, \{0,1,X,Y,B\}, \delta, q0, \{q4\})$

Estado	0	1	X	Y	B
→ qo	(q1,X,R)	--	--	(q3,Y,R)	--
q1	(q1,0,R)	(q2,Y,L)	--	(q1,Y,R)	--
q2	(q2,0,L)	--	(qo,X,R)	(q2,Y,L)	--
q3	--	--	--	(q3,Y,R)	(q4,B,R)
q4*	--	--	--	--	--

Verifique se a cadeia 000111 é aceita

Diagrama de Transição



Exercício

- Construa uma MT para reconhecer cadeias de $L=\{w\#w \mid w \in \{0,1\}^*\}$

Estágios para a resolução:

- Verifique se a entrada tem um único símbolo #, cc rejeite.
- Verifique (zigue-zague) se antes e depois do # existem os mesmos símbolos, cc rejeite. Ao checar um símbolo marque-o (use um X por exemplo) para ter controle sobre os que estão sendo analisados num dado momento.
- Quando todos os da esquerda forem checados (com X) verifique se existe algum símbolo à direita ainda não checado. Se houver, rejeite; cc aceite.

A linguagem de uma MT

- Intuitivamente: a cadeia de entrada é colocada na fita, e a cabeça da fita começa no símbolo mais à esquerda da cadeia. Se a MT entrar eventualmente num estado de aceitação, a entrada será aceita; caso contrário, não.
- Formalmente: seja M = (Q, Σ, Γ, δ, qo, F) uma MT. Então L(M) é o conjunto de cadeias w em Σ^* tais que $q_0 w \overset{*}{\vdash} \alpha p \beta$ para algum estado p em F e quaisquer cadeias de fita α e β. (aceitação por estado final)

A linguagem de uma MT

- As linguagens aceitas por MT são também chamadas de *linguagens recursivamente enumeráveis (RE)*

MT e sua parada

- Há uma outra noção de “aceitação” para MT: a aceitação por parada. Em geral, usada quando o conteúdo final da fita representa alguma resposta ao problema que a MT representa.

- Dizemos que uma MT *pára* se ela entra em um estado q, olhando um símbolo de fita X, e não existe mais nenhum movimento previsto nessa situação, i.e., $\delta(q,X)$ é indefinido.

Usos de uma MT

- como reconhecedor de linguagens (Visto)

- para calcular funções

MT como um processador de funções inteiras

- Tradicionalmente, os inteiros são representados em vocabulário unário.
- O inteiro i >= 0 é representado pela cadeia 0^{i}.
- Se a função tem k argumentos (i1, i2, ..., ik) então esses inteiros são colocados na fita separados por 1´s como:

$$0^{i1} \; 1 \; 0^{i2} \; 1 \ldots 1 \; 0^{ik}$$

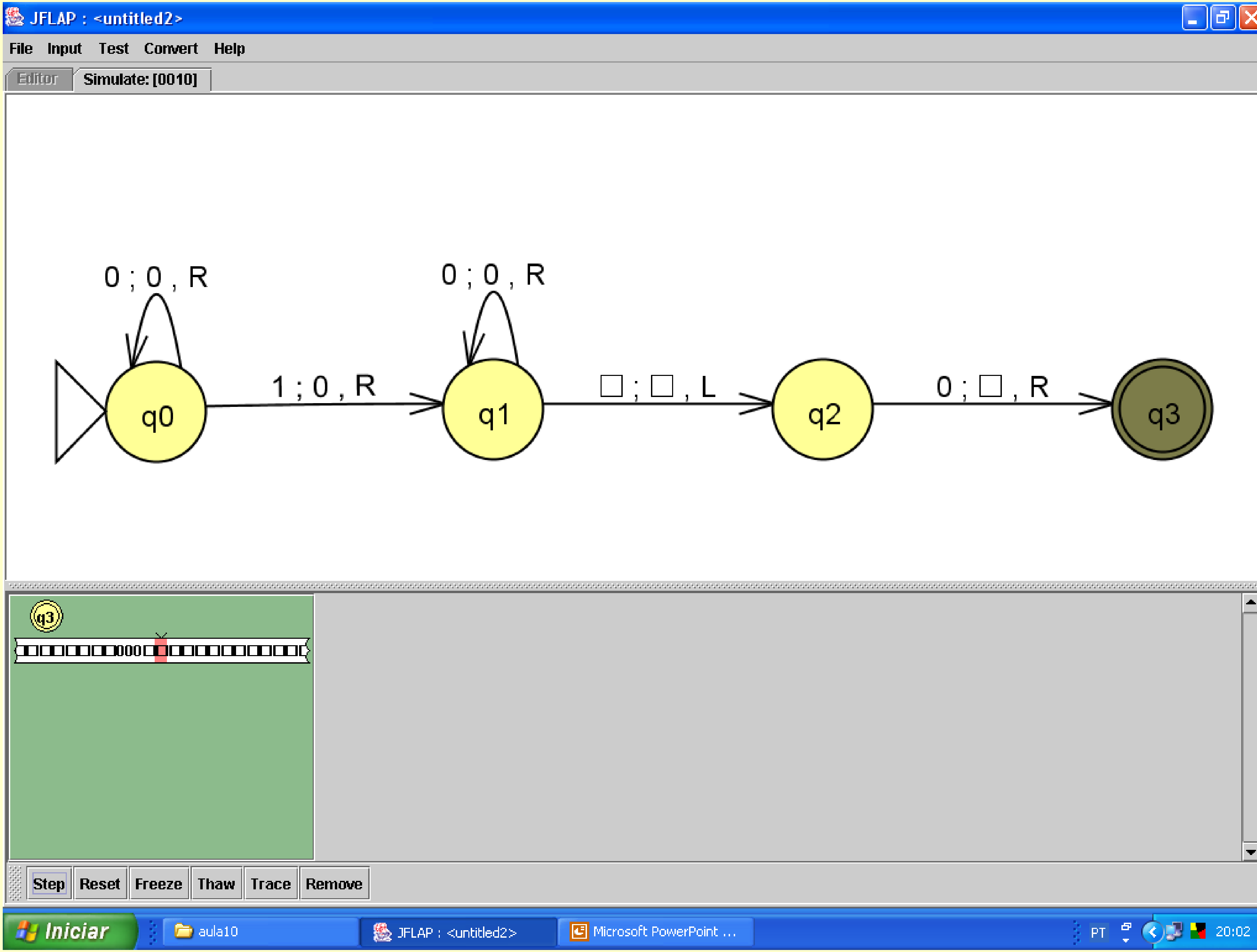
- O inverso também é possível.
- Se a máquina pára (não importa se num estado final) com a fita consistindo de 0^{m} para algum m então dizemos que f(i1,i2,...ik) = m, onde f é uma função de k argumentos computados por essa MT.

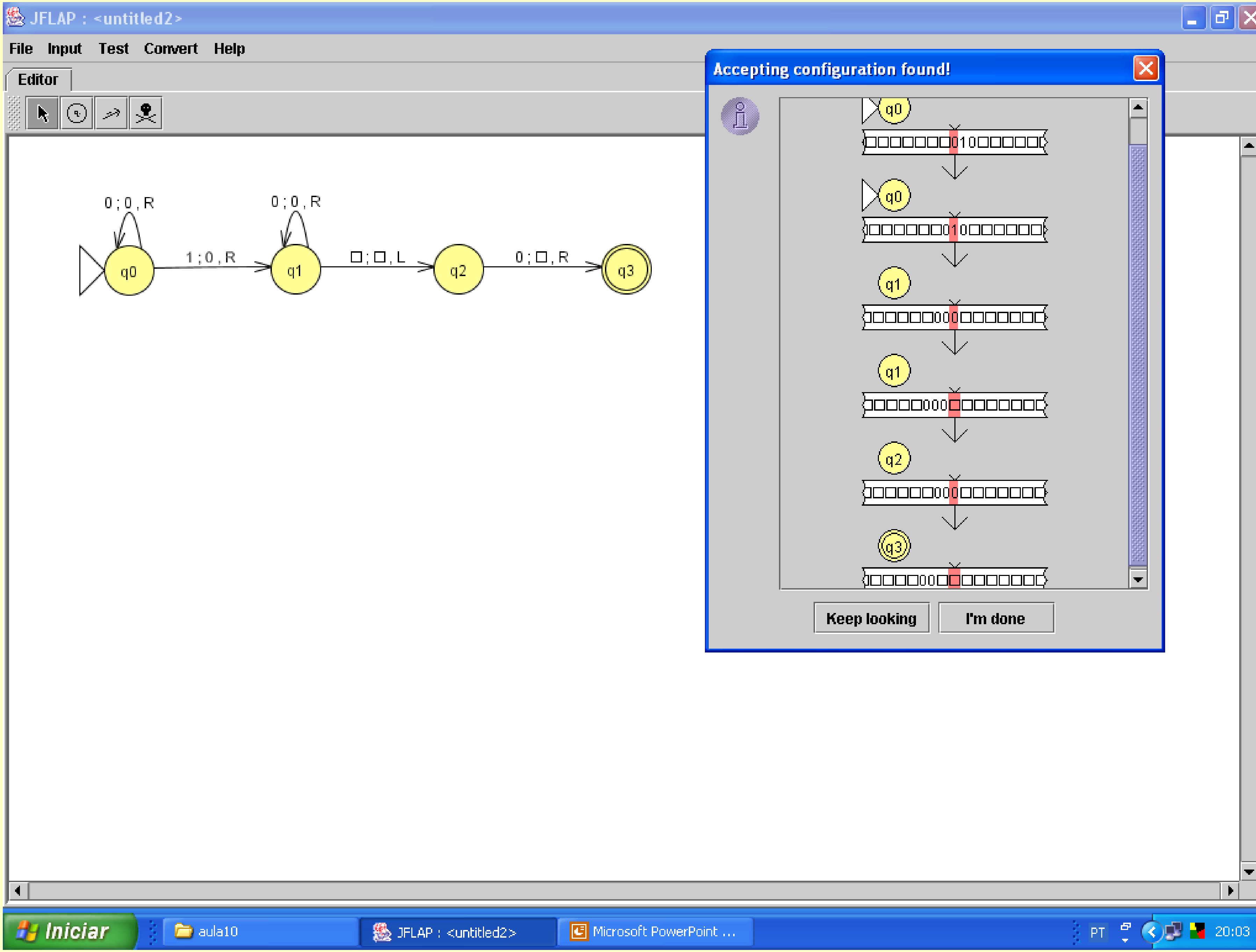
Exemplo: MT que soma dois números naturais

- Conteúdo inicial da Fita: ...B 0^a 1 0^b B...
- Quando a MT parar, o conteúdo da fita dever ser:

  ...B 0^{a+b} B....

- Processo:
- Ler o 0 mais à esquerda, mantendo-o como 0, e mover à direita até encontrar o 1.
- Substitua o 1 por 0 (nesse momento a cadeia da fita é 0^{a+b+1}. Continue movendo à direita sem mudar a fita, até que um B seja encontrado.
- Mantenha o B e mova a esquerda para encontrar o último 0 mais a direita.
- Substitua esse 0 por B. O resultado é 0^{a+b}


JFLAP : <untitled2>
File Input Test Convert Help
Editor
Simulate: [0010]
0 ; 0 , R
0 ; 0 , R
1 ; 0 , R
□ ; □ , L
0 ; □ , R
q0
q1
q2
q3
q3
000
Step Reset Freeze Thaw Trace Remove
Iniciar
aula10
JFLAP : <untitled2>
Microsoft PowerPoint ...
PT
20:02



JFLAP : <untitled2>
File Input Test Convert Help
Editor
0 ; 0 , R
0 ; 0 , R
1 ; 0 , R
□ ; □ , L
0 ; □ , R
q0
q1
q2
q3
Accepting configuration found!
Keep looking
I'm done
Iniciar
aula10
JFLAP : <untitled2>
Microsoft PowerPoint ...
PT
20:03

Exercício

- Projete uma MT que calcule, para dois inteiros positivos m e n, $m \dot{-} n$, chamada *monus* ou *subtração própria*, e definida por:

$m \dot{-} n = \max(m-n, 0)$. Isto é,

$$m \dot{-} n = \begin{cases} m-n, & \text{se } m \geq n \\ 0, & \text{se } m < n \end{cases}$$

0^n

início

...BB000.....0100.......0BB...

0^m

final

...BB000.....000.......0BB...

$0^{m \dot{-} n}$

- F é vazio se a MT é transformadora de uma cadeia de entrada em uma cadeia de saída, isto é, como um modelo para descrever procedimentos (ou computar funções).

- F é relevante quando a MT é usada para reconhecer uma linguagem.

Ex. Uma MT para reconhecer a Linguagem

$$L = \{ a^n b^n c^n \mid n \geq 0 \}$$

Exemplos:

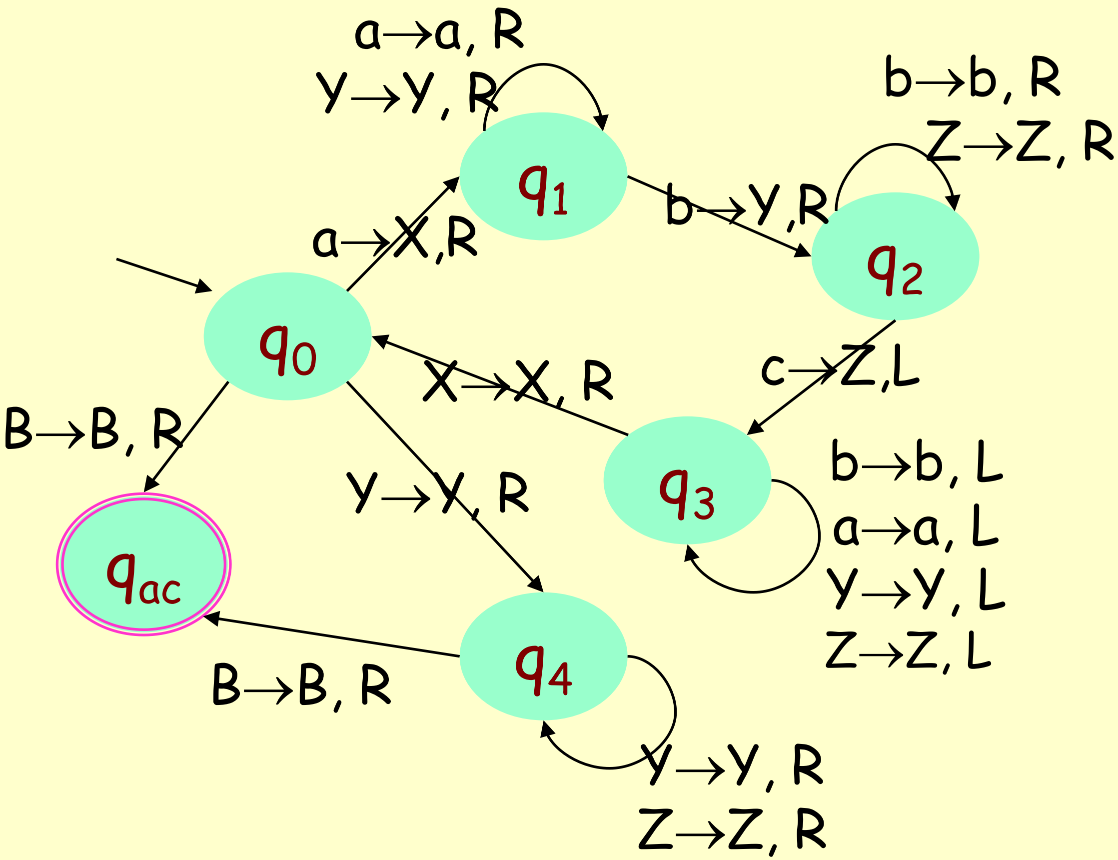
Pertence à L: aaabbbccc

Não Pertence à L: aaabbcccc

A Máquina deTuring

1. $Q = \{q_0, q_1, q_2, q_3, q_4, q_{ac}\}$
2. $\Sigma = \{a,b,c\}$
3. $\Gamma = \{a,b,c,B,X,Y,Z\}$
4. δ a seguir.
5. q_0 – o estado inicial
6. $F = \{q_{ac}\}$

Idéia: em cada passo, reconhecer um a, um b e um c.



Exercícios

1) Construir uma MT que decida se uma seqüência de parênteses é bem formada.
   - Escreve 0 se mal formada
   - Escreve 1 se bem formada

- Dica: considere que a cadeia de parênteses é limitada por 2 A´s (um a esq e outra à direita).
- Idéia: Procurem por um) e substitua por X e em seguida voltar a esquerda procurando o (mais próximo para substituir por X também.

2) Construir uma MT tal que, dada uma cadeia w pertencente ao fecho de {0,1}, duplique w. Quando a máquina parar, a fita deve conter w#w sendo que # indica fim de w.

Exercícios

1. Faça uma MT que reconheça L = {0^2^n | n >= 0} cadeias de 0 cujo tamanho é potência de 2
2. Faça uma MT que reconheça L = {x | x $\in$ {a,b,c}* e x é uma permutação de $a^n b^n c^n$ para algum n >= 0 }

   ex. aabbcc bca cccaaabbb

Comentários sobre os Exercícios

2:

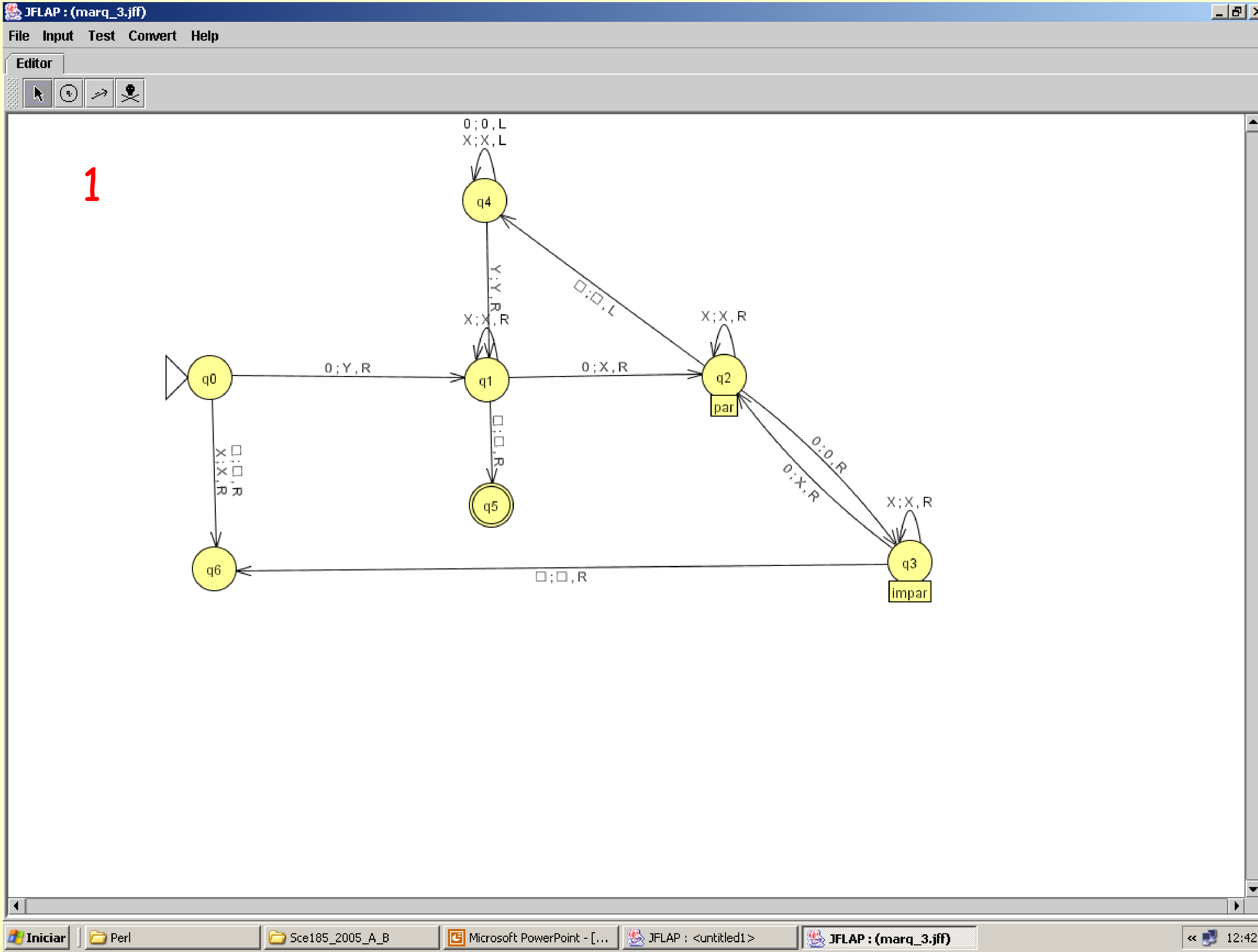
a) trocar um a,b, ou c do começo por 1 para marcar o final à esquerda;

b) substituir um a, um b e um c por 0´s.

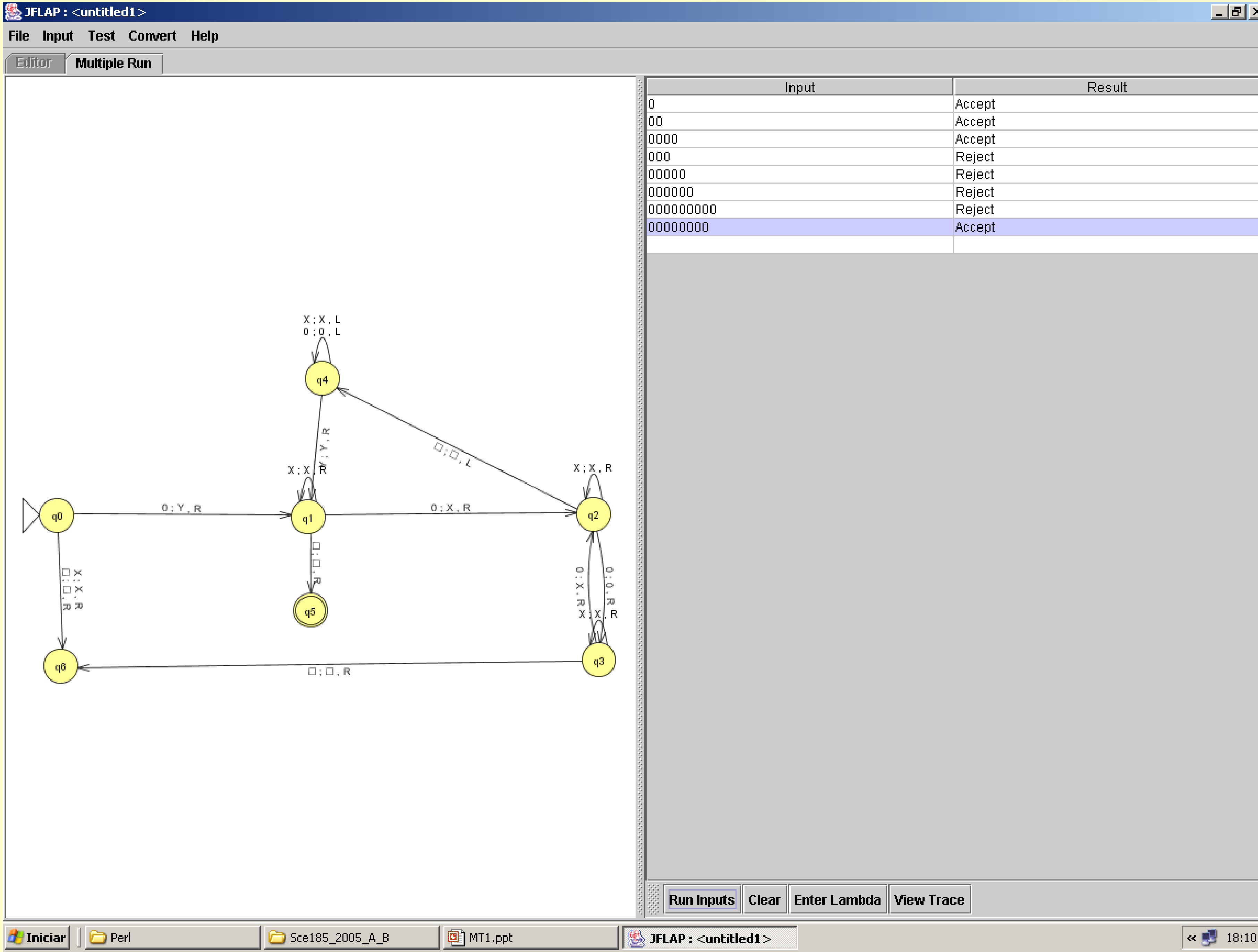
c) M aceita se, ao percorrer a cadeia de entrada, a fita consiste somente de 0´s.

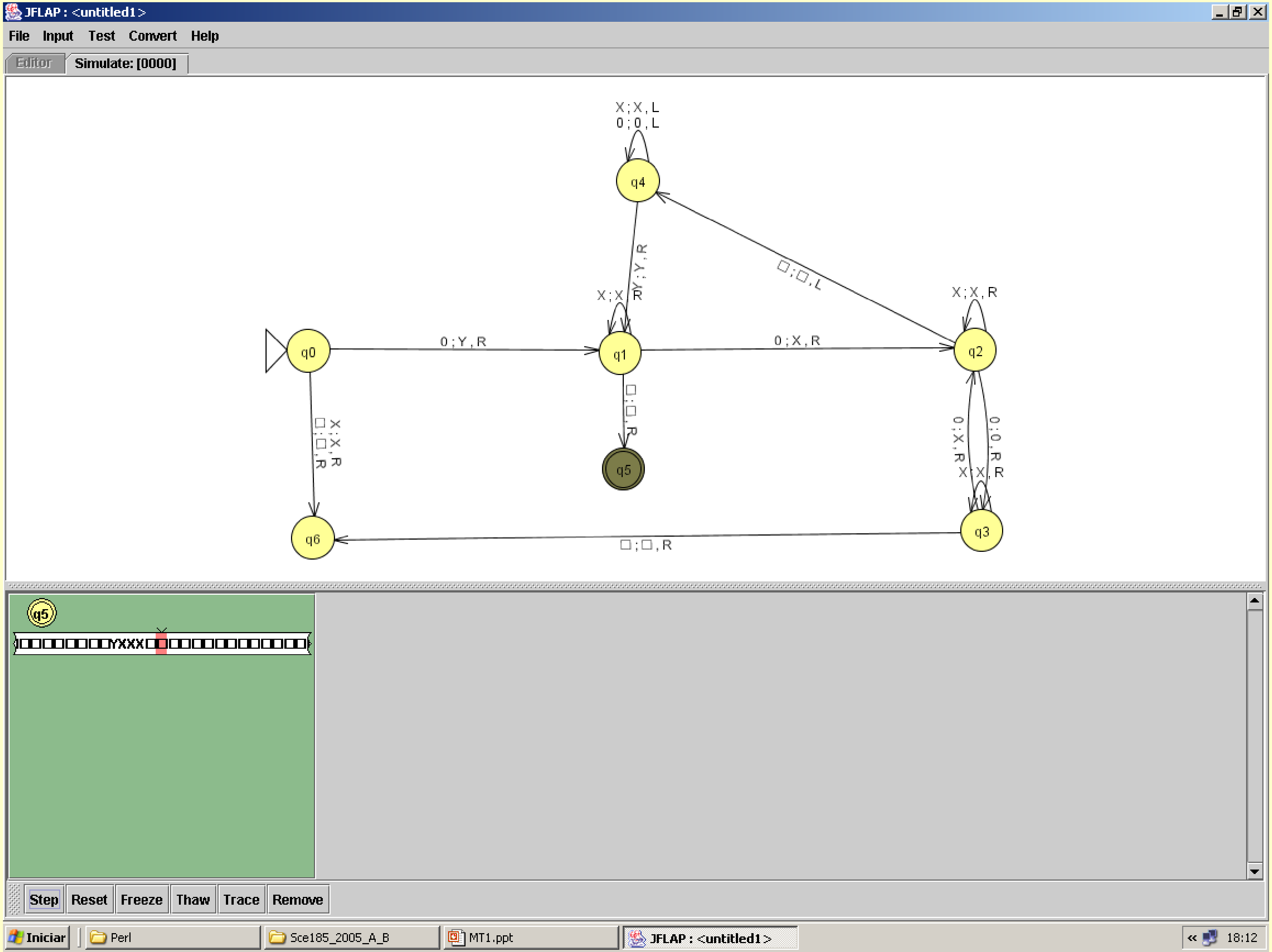
Comentários sobre os Exercícios

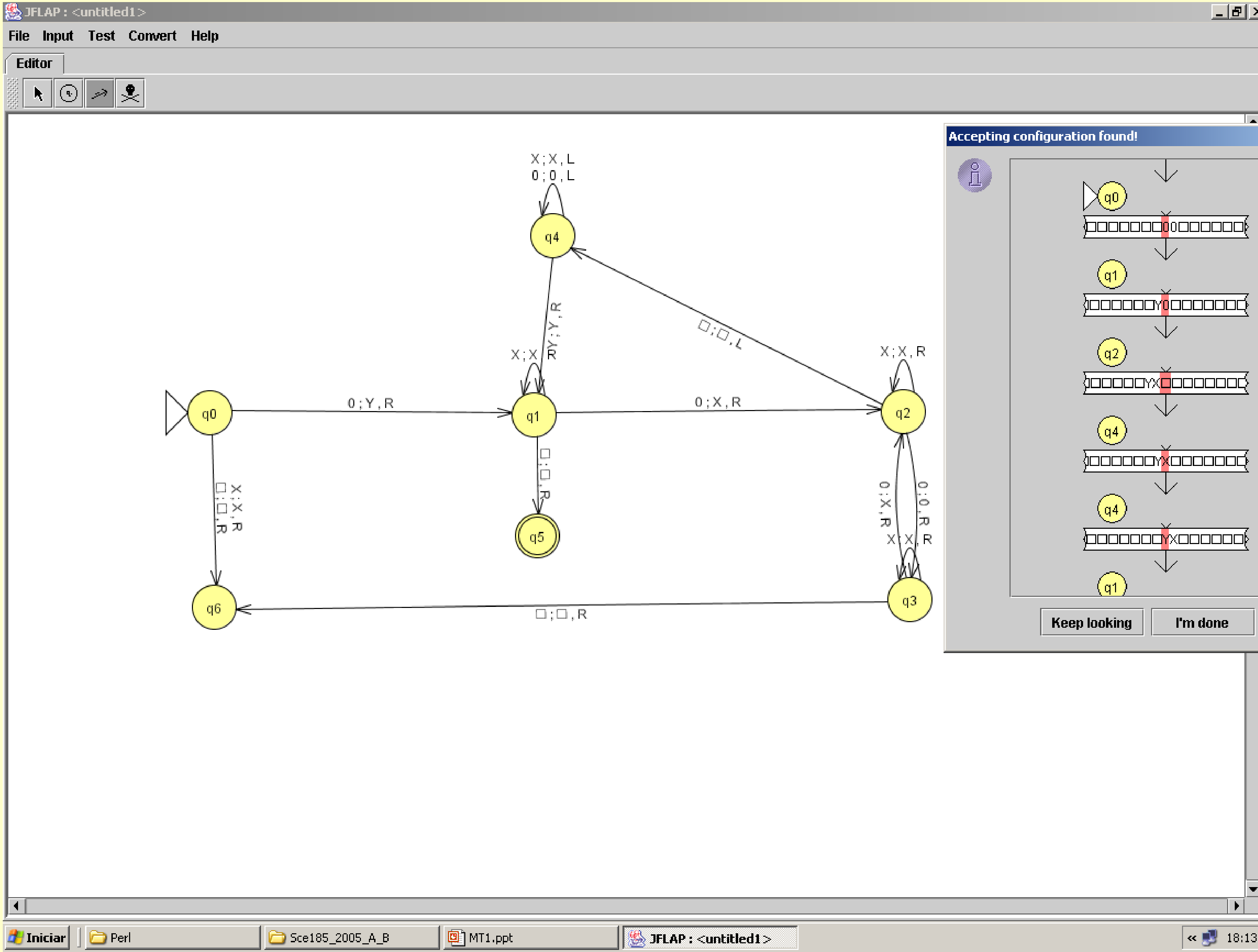
- 1: estágios para a resolução:
  0. Marque o primeiro zero com Y
  1. Atravesse da esquerda para direita marcando um zero sim outro não com um X
  2. Se no estágio 1. a fita contém 1 único 0 aceite. Se contiver mais do que 1 zero e o número for impar, rejeite.
  3. Retorne ao marcador Y
  4. Vá para o estágio 1.


JFLAP : (marq_3.jff)
File Input Test Convert Help
Editor
1
0 ; 0 , L
X ; X , L
q4
Y ; Y , R
☐ ; ☐ , L
X ; X , R
X ; X , R
q0
0 ; Y , R
q1
0 ; X , R
q2
par
0 ; 0 , R
0 ; X , R
X ; X , R
X ; X , R
☐ ; ☐ , R
☐ ; ☐ , R
q5
q6
☐ ; ☐ , R
q3
impar
Iniciar
Perl
Sce185_2005_A_B
Microsoft PowerPoint - [...
JFLAP : <untitled1>
JFLAP : (marq_3.jff)
12:42

1


JFLAP : <untitled1>
File Input Test Convert Help
Editor
Multiple Run
X ; X , L
0 ; 0 , L
q4
□ ; □ , L
X ; X , R
q0
0 ; Y , R
q1
0 ; X , R
q2
□ ; □ , R
q5
X ; X , R
0 ; X , R
0 ; 0 , R
q6
□ ; □ , R
q3
Input
Result
0
Accept
00
Accept
0000
Accept
000
Reject
00000
Reject
000000
Reject
000000000
Reject
00000000
Accept
Run Inputs
Clear
Enter Lambda
View Trace
Iniciar
Perl
Sce185_2005_A_B
MT1.ppt
JFLAP : <untitled1>
18:10


JFLAP : <untitled1>
File Input Test Convert Help
Editor Simulate: [0000]
X ; X , L
0 ; 0 , L
q4
q0
q1
q2
q3
q5
q6
0 ; Y , R
0 ; X , R
X ; X , R
□ ; □ , L
Y ; Y , R
□ ; □ , R
□ ; □ , R
X ; X , R
0 ; X , R
0 ; 0 , R
Step Reset Freeze Thaw Trace Remove
Iniciar
Perl
Sce185_2005_A_B
MT1.ppt
18:12


JFLAP : <untitled1>
File
Input
Test
Convert
Help
Editor
X ; X , L
0 ; 0 , L
q4
□ ; □ , L
X ; X
Y ; Y , R
X ; X , R
q0
0 ; Y , R
q1
0 ; X , R
q2
□ ; □ , R
X ; X , R
□ ; □ , R
q5
0 ; X , R
0 ; 0 , R
X ; X , R
q6
□ ; □ , R
q3
Accepting configuration found!
Keep looking
I'm done
Iniciar
Perl
Sce185_2005_A_B
MT1.ppt
JFLAP : <untitled1>
18:13