

# Automatos Finitos

$a^n$



Sistemas de Estados Finitos  
AF Determinísticos

## Definição formal de um AF Determinístico

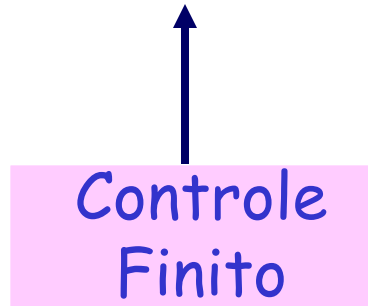
- Um AF Determinístico (AFD) é denotado formalmente por uma **quíntupla**  $(Q, \Sigma, \delta, q_0, F)$  onde:
- $Q$  é o conjunto de estados
- $\Sigma$  é um alfabeto finito (da fita)
- $q_0 \in Q$  é o estado inicial
- $F \subseteq Q$  é o conjunto de estados finais
- $\delta$  (delta) é a função de transição de estados mapeando  $Q \times \Sigma \rightarrow Q$ . **(estado, símbolo)  $\rightarrow$  estado**  
 $\delta(q_i, a) = q_j \quad \rightarrow$  **determinismo**

# Esquema da Máquina e Representação usando grafos

Esquema:



Fita com a  
sequência de  
símbolos de  $\Sigma$



## Exemplo

Seja

$L = \{w \mid w \text{ é da forma } x01y \text{ para quaisquer cadeias } x \text{ e } y \text{ que consistem somente de } 0's \text{ e } 1's\}$  ou

$$L = (0+1)^*01(0+1)^*$$

Então  $01$ ,  $11010$  e  $100011 \in L$  e

$\lambda$ ,  $0$  e  $111000 \notin L$ .

Vejam os: Seja  $A$  o autômato.

$\Sigma = \{0,1\}$ . Seja  $q_0 \in Q$  o estado inicial. Para decidir se  $01$  é uma subcadeia da entrada,  $A$  precisa lembrar:

1. Ele já viu 01? Se sim, ele aceita toda sequência de caracteres adicionais; i.e, de agora em diante ele estará sempre num estado final (de aceitação).
2. Ele nunca viu 01, mas o último caracter foi 0; assim, se agora ele vir o 1, terá visto 01 e poderá aceitar tudo que vier daqui por diante.
3. Ele nunca viu 01, mas ou ele acabou de iniciar ou acabou de encontrar 1; nesse caso, A não pode aceitar até ver primeiro um 0 seguido de 1.

Cada uma das condições acima pode ser representada por um estado. A condição (3) é o estado inicial  $q_0$ . Ficamos nesse estado até aparecer um 0. Logo  $\delta(q_0, 1) = q_0$ .

Mas se estamos em  $q_0$  e vemos um 0, estamos na condição (2). Seja  $q_2$  esse estado. Assim,  $\delta(q_0, 0) = q_2$

Se, em  $q_2$ , vemos um 0, ainda não encontramos 01, mas estamos na mesma situação de ter o 0 e esperar o 1; logo, ficamos no mesmo estado:  $\delta(q_2, 0) = q_2$ .

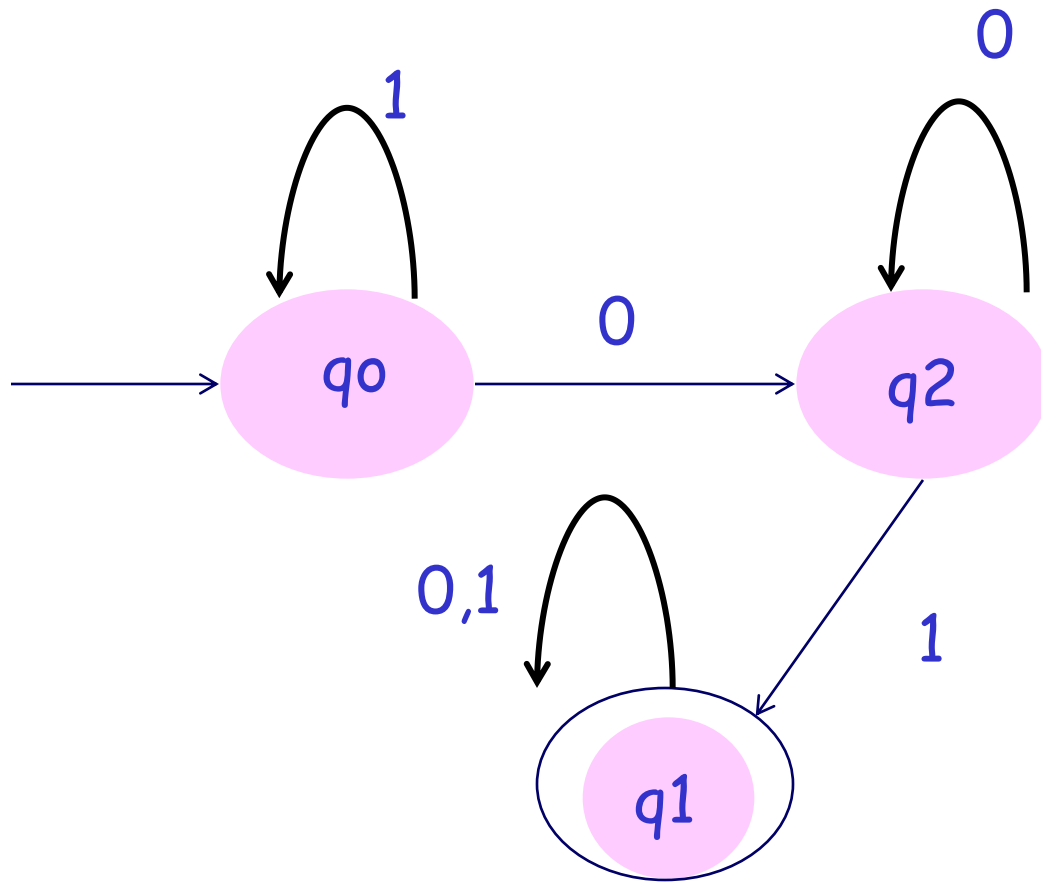
Se, no entanto, em  $q_2$ , vemos 1, então teremos a subcadeia 01 e podemos ir para um estado final, que corresponde à situação (1); chamaremos de  $q_1$ . Logo,  $\delta(q_2, 1) = q_1$ .

Em  $q_1$ , já vimos a sequência 01, então não importa o que aparecer, ainda estaremos na situação de já ter visto 01. Isto é,  $\delta(q_1, 0) = \delta(q_1, 1) = q_1$ .

Dessa forma,  $Q = \{q_0, q_1, q_2\}$ .  $q_0$  é o estado inicial e  $F = \{q_1\}$ :

$$A = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_1\}),$$

onde  $\delta$  é a função de transição descrita anteriormente.



## Aceitando cadeias - função de transição estendida

Estendemos a função de transição para aceitar cadeias  $w$ :

$$\delta': Q \times \Sigma^* \rightarrow Q$$

$\delta'(q, w)$  é um estado  $p$  onde o AF vai estar depois de ler a cadeia  $w$ , começando do estado  $q$ . Isto é, existe um caminho no diagrama de transições de  $q$  para  $p$  denominado  $w$

Definimos  $\delta'$  por indução:

1) **BASE:**  $\delta'(q, \lambda) = q$

(sem ler um símbolo de entrada o AF não pode mudar de estado)

2) **INDUÇÃO:** Suponha  $w$  uma cadeia da forma  $xa$ ; ou seja,  $a$  é o último símbolo de  $w$ , e  $x$  é a subcadeia que consiste de tudo, menos o último símbolo. Então:

$$\delta'(q, w) = \delta(\delta'(q, x), a)$$

Para calcular  $\delta'(q, w)$ , primeiro calculamos  $\delta'(q, x)$ . Suponha que esse estado seja  $p$ , ou seja,  $\delta'(q, x) = p$ . Então  $\delta'(q, w)$  é o que obtemos fazendo uma transição do estado  $p$  sobre a entrada  $a$ , o último símbolo de  $w$ . Isto é,  $\delta'(q, w) = \delta(p, a)$ .

## Linguagem aceita por um AF M

- Uma cadeia  $x$  é aceita pelo AFD  $M = (Q, \Sigma, \delta, q_0, F)$  se  $\delta'(q_0, x) = p$  para algum  $p \in F$ .

Ou

- $L(M) = \{x \mid \delta'(q_0, x) \in F\}$
- Def 1: Uma linguagem aceita por um AFD é uma Linguagem Regular
- Def 2: Dois AF  $M_1$  e  $M_2$  são equivalentes se e somente se  $L(M_1) = L(M_2)$

# Exercício

Fazer um AFD  $M$  tal que

$L(M) = \{w \mid w \in \{0,1\}^* \text{ e possui um número par de ocorrências de } 0's \text{ e de } 1's\}$

$(00+11+(01+10)(11+00)^*(10+01))^*$

<http://www.jflap.org/>

# Exemplo 1

Fazer um AFD M que aceita  $L(M) = \{w \mid w \text{ possui um nro par de } 0\text{'s e de } 1\text{'s}\}$

Diagrama de Transição de Estados

```
graph LR; q0((q0)) -- 1 --> q1((q1)); q1 -- 1 --> q0; q0 -- 0 --> q2((q2)); q2 -- 0 --> q0; q1 -- 0 --> q3((q3)); q3 -- 0 --> q1; q2 -- 1 --> q3; q3 -- 1 --> q2;
```

Cadeia aceita: configuração final VERDE

Traceback

| State | Input  |
|-------|--------|
| (q0)  | 110101 |
| q1    | 110101 |
| (q0)  | 110101 |
| q2    | 110101 |
| q3    | 110101 |
| q1    | 110101 |
| (q0)  | 110101 |

Step Reset Freeze Thaw Trace Remove

Windows Taskbar: Iniciar, SCE\_521\_185, automatos.ppt, JFLAP 4.0 Beta Ve..., JFLAP : <untitled1>, Traceback, PT, 15:44

JFLAP : (automato1)

File Input Test Convert Help

Editor Simulate: 110

```
graph LR; q0((q0)) -- 1 --> q0; q0 -- 0 --> q2((q2)); q1((q1)) -- 1 --> q0; q1 -- 0 --> q3((q3)); q2 -- 1 --> q3; q3 -- 0 --> q2;
```

N par de 0 e de par de 1

N impar de 1 e par de 0

N par de 1 e impar de 0

N impar de 0 e par de 1

q2

110

Cadeia não aceita:  
configuração final  
ROSA

Step Reset Freeze Thaw Trace Remove

Iniciat Microsoft PowerPoint - [...] JFLAP : (automato1)

11:52

## Reconhecimento de 110101

Verifique se  $\delta'(q_0, 110101) = q_0$

$$\delta(\delta'(q_0, 11010), 1) = \delta(\delta(\delta'(q_0, 1101), 0), 1) = \dots \delta(\delta \dots \delta(q_0, \lambda), 1 \dots) = q_0$$

Portanto 110101 está na  $L(M)$

$$M = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

| $\delta :$ estados             |       | símbolos |       | Função de Transição de Estados                |
|--------------------------------|-------|----------|-------|-----------------------------------------------|
|                                |       | 0        | 1     |                                               |
| Tabela de Transição de Estados | $q_0$ | $q_2$    | $q_1$ | $\delta(q_0, 0) = q_2$ $\delta(q_0, 1) = q_1$ |
|                                | $q_1$ | $q_3$    | $q_0$ | $\delta(q_1, 0) = q_3$ $\delta(q_1, 1) = q_0$ |
|                                | $q_2$ | $q_0$    | $q_3$ | $\delta(q_2, 0) = q_0$ $\delta(q_2, 1) = q_3$ |
|                                | $q_3$ | $q_1$    | $q_2$ | $\delta(q_3, 0) = q_1$ $\delta(q_3, 1) = q_2$ |

# Observações

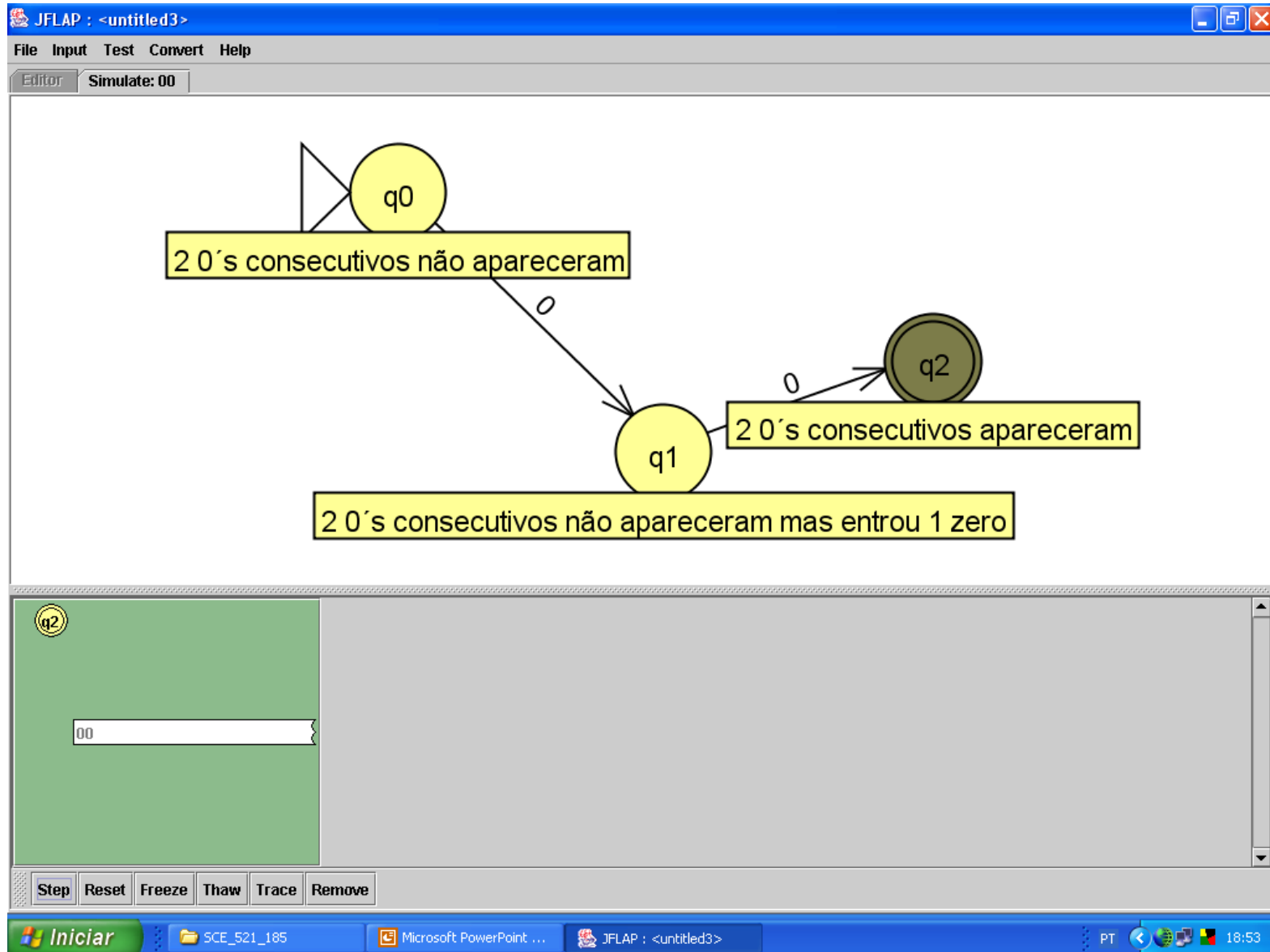
- Repare que, são os estados que servem como memória para informar a paridade dos números,
- e não a frequência dos mesmos, caso contrário precisaríamos de um número infinito de estados, para comprimentos arbitrários de cadeias.

## Exemplo 2

Fazer um AFD M que aceita

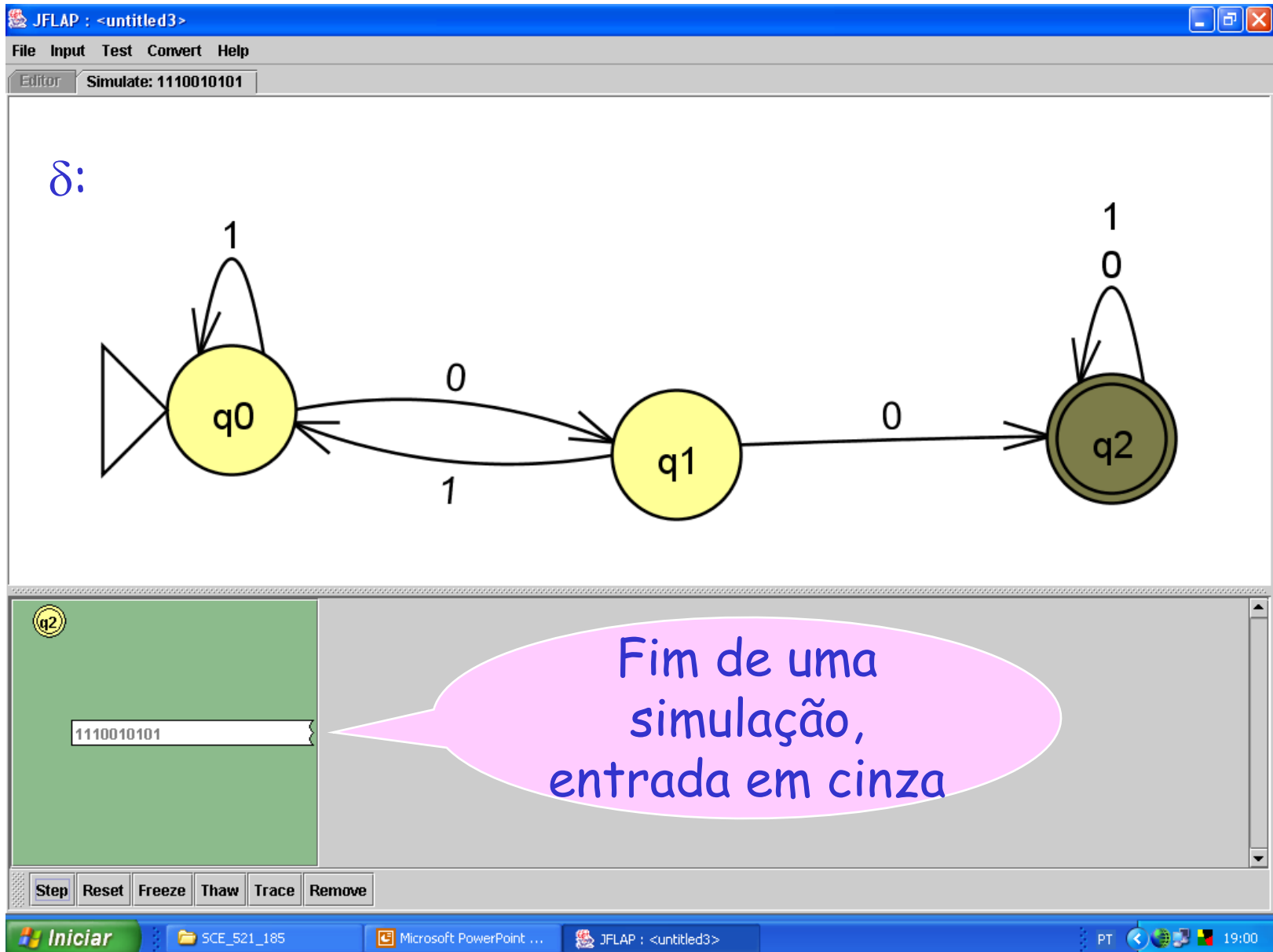
$L(M) = (0+1)^*00(0+1)^*$  = possui pelo menos dois 0's consecutivos

Garantindo  
a  
restrição...



# Completando o AFD M e reconhecendo uma cadeia de L(M)

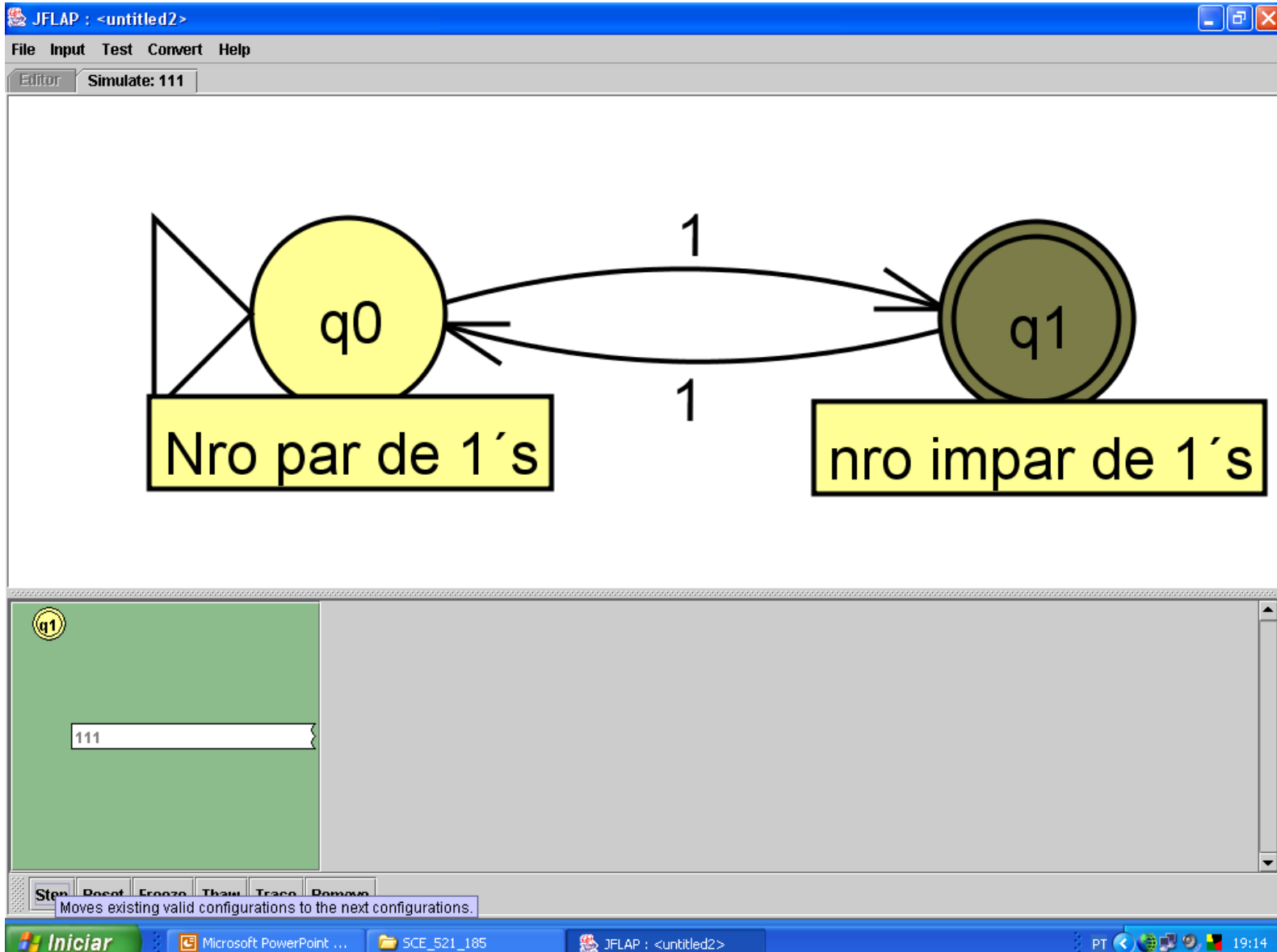
$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$$



### Exemplo 3

Fazer um AFD  $M$  que aceita  $L(M) = \{w \in \{0,1\}^* \mid w \text{ possua um número ímpar de } 1's = 0^*10^*(0^*10^*10^*)^*\}$

Garantindo  
a  
restrição...



# Completando o AFD M e reconhecendo uma cadeia de L(M)

$$M = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \{q_1\})$$

