



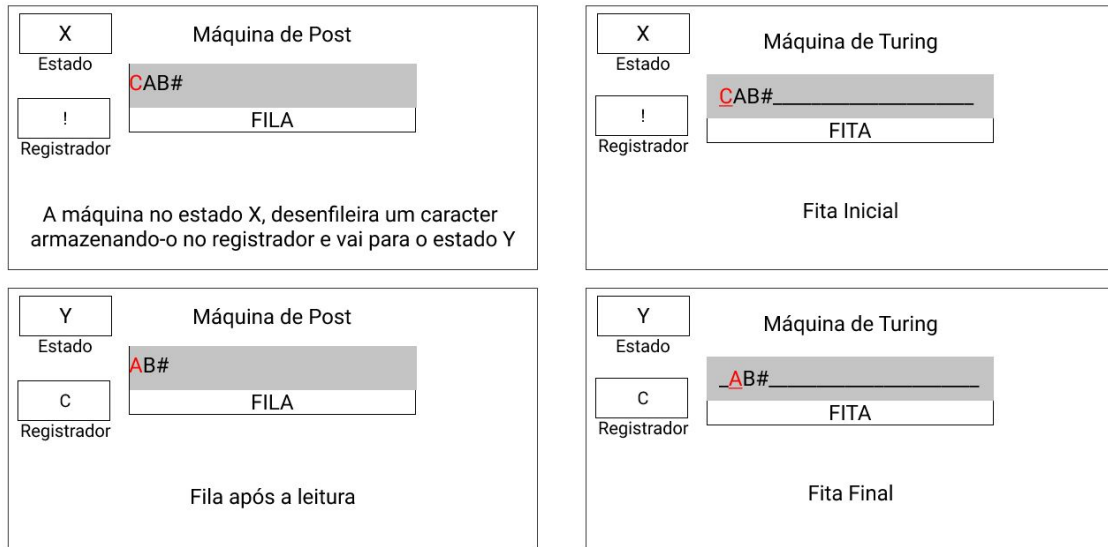
Teoria da Computação - TP1 parte 4

Simulação Máquina de Post

Professor: Wallace de Almeida Rodrigues
Aluno: Lucas Gabriel de Almeida - Número: 0035333
Aluno: Lucas Mateus Fernandes - Número: 0035411

A seguir, será demonstrado que a nossa implementação da Máquina de Post (MP) pode ser Simulada em uma Máquina de Turing versão Morphet (MT), para tal fim é necessário simular todos os comandos presentes na MP em uma MT.

Simulando o movimento 'X ler Y' na Máquina de Turing



Temos a MP, com a fila “CAB#” e sua respectiva representação na MT com a fita “CAB#”, ambos possuem o valor “!” em seu registrador e um estado arbitrário ‘X’.

Na MP ao realizar a leitura, o elemento ‘C’ é desenfileirado, com isso o registrador recebe o valor ‘C’, o estado recebe o próximo estado ‘Y’ e a fila fica como “AB#”.

Na MT para alcançar tal configuração é necessário fazer a leitura do caractere ‘C’, escrita do caractere ‘_’ e a movimentação do cabeçote para a direita, o que leva em uma fita “_AB#”, onde o ‘_’ representa o caractere vazio.

Prova:

Código inicial no Morphet

The screenshot shows the Morphet Turing Machine simulator interface. At the top, the **Tape** contains the string "CAB#" and the **Current state** is "(!,X)". A message in the center says "Machine reset. Click 'Run' or 'Step' to start." The **Stops** counter is at 0.

The **Turing machine program** is as follows:

```

1  (!,X) A _ R (A,Y)
2  (!,X) B _ R (B,Y)
3  (!,X) C _ R (C,Y)
4  (!,X) # _ R (#,Y)
5  (!,X) $ _ R ($,Y)
6
7  (A,Y) * * * halt-accept
8  (B,Y) * * * halt-accept
9  (C,Y) * * * halt-accept
10 (#,Y) * * * halt-accept
11 ($,Y) * * * halt-accept
12
13 ; 0 par ordenado representa (registrador,estado)

```

On the right, the **Controls** panel includes buttons for **Run**, **Pause**, **Step**, and **Reset**. It also has a checkbox for **Run at full speed** and an **Undo** button. The **Initial input** is set to "CAB#" and the **Initial state** is "(!,X)". The **Machine variant** is set to "Standard".

Fita após a execução do algoritmo:

The screenshot shows a Turing machine simulator interface. At the top, a yellow tape contains the string "AB#" with a red arrow pointing to the 'A'. Below the tape, the "Current state" is "halt-accept" and the machine is "Halted.". The "Turing machine program" is listed as follows:

```
1 (!,X) A _ R (A,Y)
2 (!,X) B _ R (B,Y)
3 (!,X) C _ R (C,Y)
4 (!,X) # _ R (#,Y)
5 (!,X) $ _ R ($,Y)
6
7 (A,Y) * * * halt-accept
8 (B,Y) * * * halt-accept
9 (C,Y) * * * halt-accept
10 (#,Y) * * * halt-accept
11 ($,Y) * * * halt-accept
12
13 ; 0 par ordenado representa (registrador,estado)
```

On the right, the "Controls" panel includes buttons for Run, Pause, Step, and Reset, along with an "Undo" button. It also has input fields for "Initial input" (CAB#), "Initial state" (0 X), and a "Machine variant" dropdown set to "Standard".

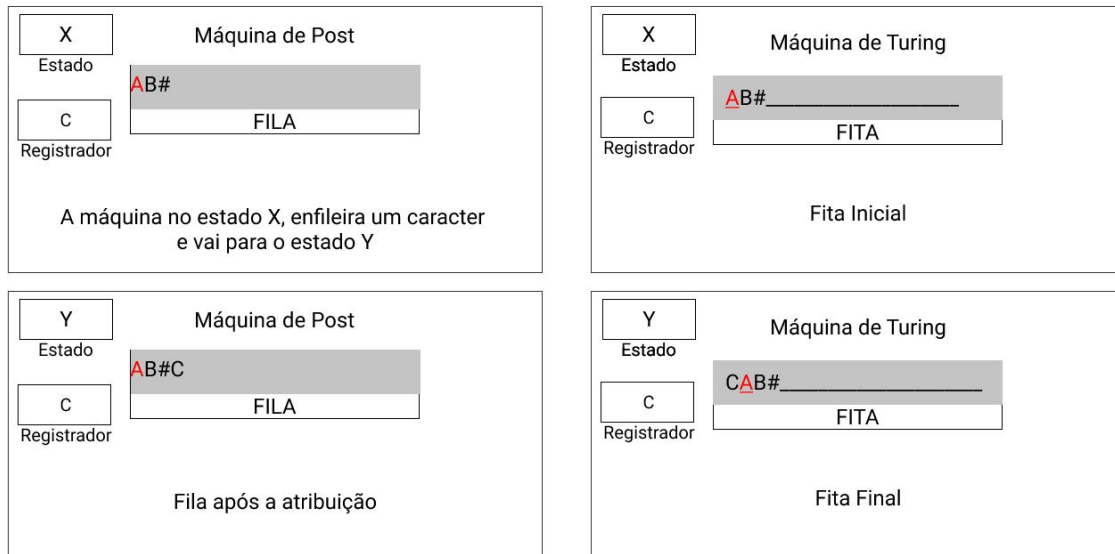
Código transcrito:

```
(!,X) A _ R (A,Y)
(!,X) B _ R (B,Y)
(!,X) C _ R (C,Y)
(!,X) # _ R (#,Y)
(!,X) $ _ R ($,Y)
```

```
(A,Y) * * * halt-accept
(B,Y) * * * halt-accept
(C,Y) * * * halt-accept
(#,Y) * * * halt-accept
($,Y) * * * halt-accept
```

```
; 0 par ordenado representa (registrador,estado)
```

Simulando o movimento 'X atrib C Y' na Máquina de Turing



Temos a MP, com a fila “AB#” e sua respectiva representação na MT com a fita “AB#”, ambos possuem o valor “C” em seu registrador e um estado arbitrário ‘X’.

Na MP ao realizar a atribuição, o elemento ‘C’ é enfileirado, fazendo com que a fita tenha “AB#C”, o registrador mantém seu valor e o estado recebe o próximo estado ‘Y’.

Caso o ‘#’ Não seja o último elemento da Fila o primeiro elemento após o ‘#’ determina o início da Fita e se não houver nenhum elemento após o ‘#’ a fila e a fita são iguais (o primeiro elemento da fita representa o primeiro elemento da fila).

A atribuição na MP adiciona ao Final da Fila o que equivale na MT adicionar o elemento antes do cabeçote de leitura.

Prova:

Código inicial do Morphett:

The screenshot shows the Morphett Turing Machine simulator interface. At the top, the **Tape** contains the string "AB#" and the **Head** is positioned at the first 'A'. The **Current state** is "tudoDireita" and the **Steps** counter is at 0. The **Machine reset. Click 'Run' or 'Step' to start.** message is displayed.

The **Turing machine program** is listed as follows:

```

1 tudoDireita _ L ultimo
2 tudoDireita * * R tudoDireita
3
4 ultimo A _ R (C,A)
5 ultimo B _ R (C,B)
6 ultimo C _ R (C,C)
7 ultimo # _ R (C,#)
8 ultimo $ _ R (C,$)
9 ultimo _ _ R (C,Y)
10
11 (C,X) A _ R (C,A)
12 (C,X) B _ R (C,B)
13 (C,X) C _ R (C,C)
14 (C,X) # _ R (C,#)
15 (C,X) $ _ R (C,$)
16
17 (C,A) * A L volta
18 (C,B) * B L volta
19 (C,C) * C L volta
20 (C,#) * # L volta
21 (C,$) * $ L volta
22
23 volta * * L ultimo
24 (C,Y) _ C R halt-accept
25
26 ; 0 par ordenado representa (registrador,estado)
27

```

The **Controls** panel on the right includes buttons for **Run**, **Pause**, **Step**, **Reset**, and **Undo**. It also features a checkbox for **Run at full speed**. The **Initial input** is set to "AB#" and the **Initial state** is "tudoDire". The **Machine variant** is set to "Standard".

Fita após a execução do algoritmo:

The screenshot shows a Turing machine simulation interface. At the top, the tape is represented as a yellow bar with the content "C A B #". Below the tape, the current state is "halt-accept" and the machine is "Halted.". The number of steps taken is 15. The Turing machine program is listed as follows:

```
1 tudoDireita _ _ L ultimo
2 tudoDireita * * R tudoDireita
3
4 ultimo A _ R (C,A)
5 ultimo B _ R (C,B)
6 ultimo C _ R (C,C)
7 ultimo # _ R (C,#)
8 ultimo $ _ R (C,$)
9 ultimo _ _ R (C,Y)
10
11 (C,X) A _ R (C,A)
12 (C,X) B _ R (C,B)
13 (C,X) C _ R (C,C)
14 (C,X) # _ R (C,#)
15 (C,X) $ _ R (C,$)
16
17 (C,A) * A L volta
18 (C,B) * B L volta
19 (C,C) * C L volta
20 (C,#) * # L volta
21 (C,$) * $ L volta
22
23 volta * * L ultimo
24
25 (C,Y) _ C R halt-accept
26
27 ; 0 par ordenado representa (registrador,estado)
```

On the right side, there is a "Controls" panel with buttons for "Run", "Pause", "Step", "Reset", and "Undo". It also includes fields for "Initial input: AB#", "Initial state: tudoDire", and a "Machine variant" dropdown set to "Standard".

Código transcrito:

```
tudoDireita _ _ L ultimo
tudoDireita * * R tudoDireita
```

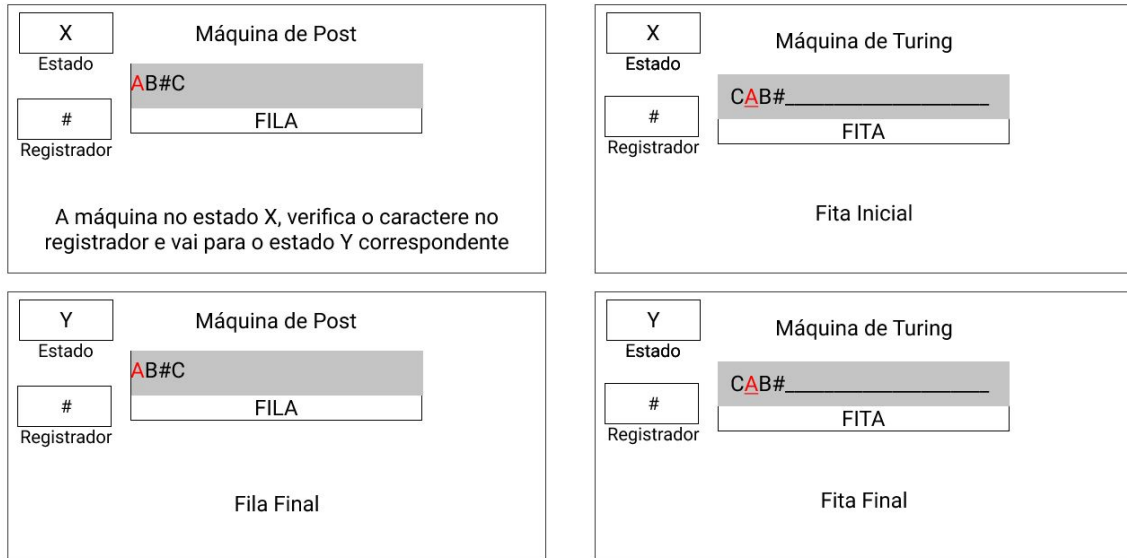
```
ultimo A _ R (C,A)
ultimo B _ R (C,B)
ultimo C _ R (C,C)
ultimo # _ R (C,#)
ultimo $ _ R (C,$)
ultimo _ _ R (C,Y)
```

```
(C,X) A _ R (C,A)
(C,X) B _ R (C,B)
(C,X) C _ R (C,C)
(C,X) # _ R (C,#)
(C,X) $ _ R (C,$)
```

```
(C,A) * A L volta
(C,B) * B L volta
(C,C) * C L volta
(C,#) * # L volta
(C,$) * $ L volta
```

```
volta * * L ultimo
(C,Y) _ C R halt-accept
; 0 par ordenado representa (registrador,estado)
```

Simulando o movimento 'X if # Y' na Máquina de Turing



Temos a MP, com a fita “AB#C” e sua respectiva representação na MT com a fita “CAB#”, ambos possuem o valor “#” em seu registrador e um estado arbitrário ‘X’.

Na MP ao realizar a condicional, o estado é alterado caso o argumento seja igual ao valor do registrador .

A MT deve verificar o registrador e trocar apenas o estado sem alterar o conteúdo da fita e seu registrador, porém é permitido que passe por estados intermediários que sirvam de controle que irão definir se a troca de estado X->Y ocorrerá ou não.

Prova

Código inicial do Morphet:

The screenshot shows the Morphet Turing Machine simulator interface. At the top, the tape (Tape) contains 'CAB#' and the head (Head) is positioned over the 'A'. The current state (Current state) is 'inicio' and the number of steps (Steps) is 0. The machine is reset, and the user is prompted to click 'Run' or 'Step' to start.

The Turing machine program (Turing machine program) is displayed in a text area:

```

1 inicio * * R inicio'
2 inicio' * * * (Reg,X)
3 (Reg,X) * * * (#,X)
4
5 (#,X) A # * (A,X')
6 (#,X) B # * (B,X')
7 (#,X) C # * (C,X')
8 (#,X) # # * (#,X')
9 (#,X) $ # * ($,X')
10
11 ;Substituir o primeiro * por cada letra do alfabeto
12 (*,X) A * * (A,X')
13 (*,X) B * * (B,X')
14 (*,X) C * * (C,X')
15 (*,X) # * * (#,X')
16 (*,X) $ * * ($,X')
17
18 (A,X') # A * (#,Y)
19 (A,X') * A * halt-reject
20
21 (B,X') # B * (#,Y)
22 (B,X') * B * halt-reject
23
24 (C,X') # C * (#,Y)
25 (C,X') * C * halt-reject
26
27 (#,X') # # * (#,Y)
28 (#,X') * # * halt-reject

```

The Controls panel on the right includes buttons for Run, Pause, Step, and Reset. It also has a checkbox for 'Run at full speed' and an 'Undo' button. The initial input is 'CAB#' and the initial state is 'inicio'. The machine variant is set to 'Standard'.

Fita após a execução do algoritmo:



Código transcrito:

```

inicio * * R inicio'
inicio' * * * (Reg,X)
(Reg,X) * * * (#,X)

```

```

(#,X) A # * (A,X')
(#,X) B # * (B,X')
(#,X) C # * (C,X')
(#,X) # # * (#,X')
(#,X) $ # * ($,X')

```

```

;Substituir o primeiro * por cada letra do alfabeto
(*,X) A * * (A,X')
(*,X) B * * (B,X')
(*,X) C * * (C,X')
(*,X) # * * (#,X')
(*,X) $ * * ($,X')

```

```

(A,X') # A * (#,Y)
(A,X') * A * halt-reject

```

```

(B,X') # B * (#,Y)
(B,X') * B * halt-reject

```

```

(C,X') # C * (#,Y)
(C,X') * C * halt-reject

```

```

(#,X') # # * (#,Y)

```

$(\#,X') * \# * \text{halt-reject}$

$(\$,X') \# \$ * (\#,Y)$

$(\$,X') * \$ * \text{halt-reject}$

$(\#,Y) * * * \text{halt-accept}$

; O par ordenado representa (registrador,estado)