

ECM253 – Linguagens Formais, Autômatos e Compiladores

Exemplo detalhado

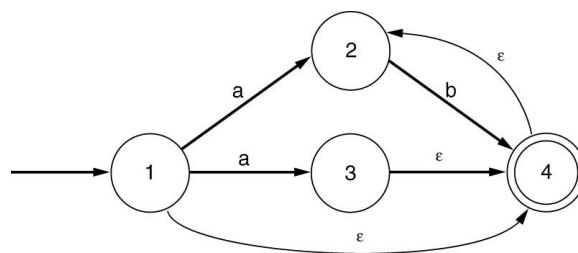
Algoritmos de um scanner

Marco Furlan

Agosto/2018

1 Problema

Considerar o autômato não determinístico apresentado na figura a seguir:



Pede-se:

- Aplicar o algoritmo de para calcular o fechamento- ϵ e determinar o fechamento fechamento- ϵ de cada estado do autômato apresentado.
- Com o resultado obtido no item anterior, aplicar o algoritmo de construção por subconjunto para determinar os estados do autômato determinístico equivalente.

- (c) Aplicar o algoritmo de Hopcroft para minimizar o autômato não determinístico obtido anteriormente. Desenhar o autômato obtido.

2 Solução

(a)

Algorithm 1 Algoritmo para calcular o fechamento- ϵ

```

1: for cada estado  $n \in N$  do                                 $\triangleright N$  é o conjunto de estados
2:    $E(n) \leftarrow \{n\}$                                       $\triangleright E(n)$  é o fechamento- $\epsilon$  de  $n$  – a resposta desejada
3: end for
4:  $WorkList \leftarrow N$ 
5: while  $WorkList \neq \emptyset$  do
6:   remover  $n$  de  $WorkList$ 
7:    $t \leftarrow \{n\} \cup \bigcup_{n \xrightarrow{\epsilon} p \in \delta_N} E(p)$            $\triangleright$  formar em  $t$  um conjunto de todos
                                                                 $\triangleright$  os estados alcançáveis por transições  $\epsilon$ 
8:   if  $t \neq E(n)$  then
9:      $E(n) \leftarrow t$ 
10:     $WorkList \leftarrow WorkList \cup \{m \mid m \xrightarrow{\epsilon} n \in \delta_N\}$   $\triangleright$  adiciona os estados  $m$ 
                                                                que possuem
                                                                 $\triangleright$  transição  $\epsilon$  para o estado  $n$  em questão
11:   end if
12: end while

```

Simulação

linhas 1–4 Calcula-se $E(n)$ e $Worklist$ assim:

Elemento	Valor
$E(1)$	$\{1\}$
$E(2)$	$\{2\}$
$E(3)$	$\{3\}$
$E(4)$	$\{4\}$
$WorkList$	$\{1,2,3,4\}$

linha 5 – iteração 1 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	{1}
$E(2)$	{2}
$E(3)$	{3}
$E(4)$	{4}
n	1
t	{1, 4}
$WorkList$	{2,3,4}

linhas 8 O teste é verdadeiro

linhas 9–10 Atualiza $E(1)$ e $WorkList$:

Elemento	Valor
$E(1)$	{1,4}
$E(2)$	{2}
$E(3)$	{3}
$E(4)$	{4}
n	1
t	{1, 4}
$WorkList$	{2,3,4}

linha 5 – iteração 2 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	{1, 4}
$E(2)$	{2}
$E(3)$	{3}
$E(4)$	{4}
n	2
t	{2}
$WorkList$	{3,4}

linhas 8 O teste é falso

linha 5 – iteração 3 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	{1, 4}
$E(2)$	{2}
$E(3)$	{3}
$E(4)$	{4}
n	3
t	{3,4}
$WorkList$	{4}

linhas 8 O teste é verdadeiro

linhas 9–10 Atualiza $E(3)$ e $WorkList$:

Elemento	Valor
$E(1)$	$\{1,4\}$
$E(2)$	$\{2\}$
$E(3)$	$\{3, 4\}$
$E(4)$	$\{4\}$
n	3
t	$\{3, 4\}$
$WorkList$	$\{4\}$

linha 5 – iteração 4 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	$\{1, 4\}$
$E(2)$	$\{2\}$
$E(3)$	$\{3, 4\}$
$E(4)$	$\{4\}$
n	4
t	$\{2,4\}$
$WorkList$	\emptyset

linhas 8 O teste é verdadeiro

linhas 9–10 Atualiza $E(4)$ e $WorkList$:

Elemento	Valor
$E(1)$	$\{1, 4\}$
$E(2)$	$\{2\}$
$E(3)$	$\{3, 4\}$
$E(4)$	$\{2, 4\}$
n	4
t	$\{2,4\}$
$WorkList$	$\{1,3\}$

linha 5 – iteração 5 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	$\{1, 4\}$
$E(2)$	$\{2\}$
$E(3)$	$\{3, 4\}$
$E(4)$	$\{2, 4\}$
n	1
t	$\{1, 2, 4\}$
$WorkList$	$\{3\}$

linhas 8 O teste é verdadeiro

linhas 9–10 Atualiza $E(1)$ e $WorkList$:

Elemento	Valor
$E(1)$	{1, 2, 4}
$E(2)$	{2}
$E(3)$	{3, 4}
$E(4)$	{2, 4}
n	1
t	{1, 2, 4}
$WorkList$	{3}

linha 5 – iteração 5 O teste é verdadeiro

linhas 6–7 Atualiza n , t e $WorkList$:

Elemento	Valor
$E(1)$	{1, 2, 4}
$E(2)$	{2}
$E(3)$	{3, 4}
$E(4)$	{2, 4}
n	3
t	{2, 3, 4}
$WorkList$	\emptyset

linhas 8 O teste é verdadeiro

linhas 9–10 Atualiza $E(3)$ e $WorkList$:

Elemento	Valor
$E(1)$	{1, 2, 4}
$E(2)$	{2}
$E(3)$	{2, 3, 4}
$E(4)$	{2, 4}
n	3
t	{2, 3, 4}
$WorkList$	\emptyset

linha 5 O teste é falso. Sair do laço. O resultado é:

Elemento	Valor
$E(1)$	{1, 2, 4}
$E(2)$	{2}
$E(3)$	{2, 3, 4}
$E(4)$	{2, 4}

(b)

Algorithm 2 Algoritmo de construção de subconjuntos

```
1:  $q_0 \leftarrow E(n_0)$  ▷  $E(n)$  é calculado pelo algoritmo anterior
2:  $Q \leftarrow \{q_0\}$ 
3:  $WorkList \leftarrow \{q_0\}$ 
4: while  $WorkList \neq \emptyset$  do
5:   remover  $q$  de  $WorkList$ 
6:   for cada caractere  $c \in \Sigma$  do
7:      $t \leftarrow E(\delta(q, c))$  ▷ próximo estado
8:      $T[q, c] \leftarrow t$  ▷ atualiza a tabela de estados final
9:     if  $t \notin Q$  then
10:       adicione  $t$  a  $Q$  e também a  $WorkList$ 
11:     end if
12:   end for
13: end while
```

Simulação**linhas 1–3** Valores iniciais

Elemento	Valor
q_0	$\{1, 2, 4\}$
Q	$\{\{1, 2, 4\}\}$
$WorkList$	$\{\{1, 2, 4\}\}$

linha 4 – iteração 1 O teste é verdadeiro**linha 5** Atualiza q e $WorkList$:

Elemento	Valor
q	$\{1, 2, 4\}$
Q	$\{\{1, 2, 4\}\}$
$WorkList$	\emptyset

linhas 6–12 Atualiza T , Q e $WorkList$:

Elemento	Valor
q	$\{1, 2, 4\}$
$T[\{1, 2, 4\}, a]$	$\{\{2\}\}$
$T[\{1, 2, 4\}, b]$	$\{\{2, 4\}\}$
Q	$\{\{1, 2, 4\}, \{2\}, \{2, 4\}\}$
$WorkList$	$\{\{2\}, \{2, 4\}\}$

linha 4 – iteração 2 O teste é verdadeiro**linha 5** Atualiza q e $WorkList$:

Elemento	Valor
q	$\{2\}$
Q	$\{\{1, 2, 4\}, \{2\}, \{2, 4\}\}$
$WorkList$	$\{\{2, 4\}\}$

linhas 6–12 Atualiza T , Q e $WorkList$:

Elemento	Valor
q	$\{2\}$
$T[\{1, 2, 4\}, a]$	$\{\{2\}\}$
$T[\{1, 2, 4\}, b]$	$\{\{2, 4\}\}$
$T[\{2\}, a]$	\emptyset
$T[\{2\}, b]$	$\{\{2, 4\}\}$
Q	$\{\{1, 2, 4\}, \{2\}, \{2, 4\}\}$
$WorkList$	$\{\{2, 4\}\}$

linha 4 – iteração 3 O teste é verdadeiro

linha 5 Atualiza q e $WorkList$:

Elemento	Valor
q	$\{2, 4\}$
Q	$\{\{1, 2, 4\}, \{2\}, \{2, 4\}\}$
$WorkList$	\emptyset

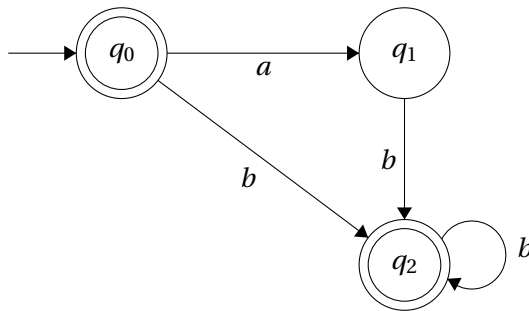
linhas 6–12 Atualiza T , Q e $WorkList$:

Elemento	Valor
q	$\{2, 4\}$
$T[\{1, 2, 4\}, a]$	$\{\{2\}\}$
$T[\{1, 2, 4\}, b]$	$\{\{2, 4\}\}$
$T[\{2\}, a]$	\emptyset
$T[\{2\}, b]$	$\{\{2, 4\}\}$
$T[\{2, 4\}, a]$	\emptyset
$T[\{2, 4\}, b]$	$\{\{2, 4\}\}$
Q	$\{\{1, 2, 4\}, \{2\}, \{2, 4\}\}$
$WorkList$	\emptyset

linha 4 O teste é falso. O resultado está na tabela de transição para o DFA equivalente:

Elemento	Valor
$T[\{1, 2, 4\}, a]$	$\{\{2\}\}$
$T[\{1, 2, 4\}, b]$	$\{\{2, 4\}\}$
$T[\{2\}, a]$	\emptyset
$T[\{2\}, b]$	$\{\{2, 4\}\}$
$T[\{2, 4\}, a]$	\emptyset
$T[\{2, 4\}, b]$	$\{\{2, 4\}\}$

Denominando $q_0 = \{1, 2, 4\}$, $q_1 = \{2\}$ e $q_2 = \{2, 4\}$, tem-se o DFA:



(c)

Algorithm 3 Algoritmo de separação (*split*) de partição

```

1: function SPLIT( $S$ )  $\triangleright S$  é um conjunto de conjuntos que define uma partição
   de estados
2:   for cada  $c \in \Sigma$  do
3:     if  $c$  separa  $S$  em  $s_1$  e  $s_2$  then
4:       return  $\{s_1, s_2\}$ 
5:     end if
6:   end for
7:   return  $S$   $\triangleright$  se não houve separação retorna  $S$  original
8: end function

```

Nota: Se o símbolo c separa S em $\{s_1, s_2\}$ é porque existe um rótulo c na transição de um ou mais estados de S de modo seu(s) próximo(s) estados não estão na mesma partição que os demais. Daí o algoritmo retorna um conjunto de dois conjuntos de estados no lugar de S .

Algorithm 4 Algoritmo de Hopcroft

```

1:  $T \leftarrow \{D_A, D - D_A\}$ 
2:  $P \leftarrow \emptyset$ 
3: while  $P \neq T$  do
4:    $P \leftarrow T$ 
5:    $T \leftarrow \emptyset$ 
6:   for cada conjunto  $p \in P$  do
7:      $T \leftarrow T \cup \text{Split}(p)$ 
8:   end for
9: end while

```

Simulação: utiliza o DFA do item anterior.

linhas 1–2 Inicializar T e P

Elemento	Valor
T	$\{\{q_0, q_2\}, \{q_1\}\}$
P	\emptyset

linha 3 - iteração 1 O teste é verdadeiro

linhas 4–5 Atualizar P e T

Elemento	Valor
T	\emptyset
P	$\{\{q_0, q_2\}, \{q_1\}\}$

linhas 6–8 – iteração 1 Atualizar T

Elemento	Valor
T	$\{\{q_0\}, \{q_2\}\}$
p	$\{q_0, q_2\}$
P	$\{\{q_0, q_2\}, \{q_1\}\}$

linhas 6–8 – iteração 2 Atualizar T

Elemento	Valor
T	$\{\{q_0\}, \{q_1\}, \{q_2\}\}$
p	$\{q_1\}$
P	$\{\{q_0, q_2\}, \{q_1\}\}$

linha 3 - iteração 2 O teste é verdadeiro

linhas 4–5 Atualizar P e T

Elemento	Valor
T	\emptyset
P	$\{\{q_0\}, \{q_1\}, \{q_2\}\}$

linhas 6–8 – iterações 1, 2 e 3 Atualizar T

Elemento	Valor
T	$\{\{q_0\}, \{q_1\}, \{q_2\}\}$
P	$\{\{q_0\}, \{q_1\}, \{q_2\}\}$

linha 3 O teste é falso. O DFA já estava minimizado pois não houve alteração nas partições iniciais.