Element of the Theory of the Computation

Lecture 1

Sets: Example 1

- Decide whether the following statements are true or not!
  - $5 \in \{1, 2, 3, 4, 5\}$
  - $0 \in \{1, 2, 3, 4, 5\}$
  - $\{2, 3\} \in \{\{1, 3\}, 2, \{2, 3, 4\}, \{2, 3, \emptyset\}\}$
  - $\emptyset \in \{1, 2, 3, 4, 5\}$
  - $7 \in \{1, 3, 5, 7\}$
  - $1 \in \{\{1\}, 2, 3, 5, 7\}$
  - $\{2, 3\} \in \{1, 2, \{3, 2\}, 4\}$
  - $\{1, 2\} \in \{1, 2, \{2, 3\}\}$
  - $\emptyset \in \{\emptyset, 1, \{2, 4\}\}$

Sets: Example 2

- Decide whether the following statements are true or not!
  - $\{1, 2\} \subseteq \{2, 1\}$
  - $\{\{1\}\} \subseteq \{1, 2\}$
  - $\{1\} \subseteq \{1, 2\}$
  - $\{1\} \subseteq \{3, 2\}$
  - $\emptyset \subseteq \{1, 2\}$
  - $\{2, 3\} \subseteq \{1, 3, \{2, 4\}\}$
  - $\emptyset \subseteq \{\emptyset, 3, \{3, 4\}\}$
  - $\{\{2, 3\}\} \subseteq \{1, 3, \{2, 3\}\}$

Sets: Example 3

- Decide whether the following statements are true or not!
  - $\emptyset \subseteq \emptyset$
  - $\emptyset \in \emptyset$
  - $\emptyset \subseteq \{\emptyset\}$
  - $\emptyset \in \{\emptyset\}$
  - $\{a, b\} \subseteq \{a, b, c\}$
  - $\{a, b\} \in \{a, b\}$
  - $\{a, b\} \subseteq \{a, b, \{a, b\}\}$
  - $\{\{a, b\}\} \subseteq \{a, b, \{a, b\}\}$
  - $\{a, b\} \in \{a, b, \{b, a\}\}$

Power sets: Example 1

- Which elements are inside the power set?
  - P({a, b}) =

0: excluded, 1: included

a	b	P({a, b})
0	0	$\emptyset$
0	1	{b}
1	0	{a}
1	1	{a, b}

  - P({a, b}) = {$\emptyset$, {a}, {b}, {a, b}}

Power sets: Example 2

- Which elements are inside the power sets?
  - $2^{\{5,\,7\}} =$

0: excluded, 1: included

5	7	P({5, 7})
0	0	$\emptyset$
0	1	{7}
1	0	{5}
1	1	{5, 7}

  - $2^{\{5,\,7\}} = \{\emptyset, \{5\}, \{7\}, \{5, 7\}\}$

Power sets: Example 3

- Which elements are inside the power sets?
  - $P(\{2, 3, 5\})$ =

2	3	5	$P(\{2, 3, 5\})$
0	0	0	$\varnothing$
0	0	1	$\{5\}$
0	1	0	$\{3\}$
0	1	1	$\{3, 5\}$
1	0	0	$\{2\}$
1	0	1	$\{2, 5\}$
1	1	0	$\{2, 3\}$
1	1	1	$\{2, 3, 5\}$

  - $P(\{2, 3, 5\}) = \{\varnothing, \{2\}, \{3\}, \{5\}, \{2, 3\}, \{2, 5\}, \{3, 5\}, \{2, 3, 5\}\}$

Power sets: Example 4

- Which elements are inside the power sets?
  - P({x}) =

0: excluded, 1: included

x	P({x})
0	{}
1	{x}

  - P({x}) = {{}, {x}}

Power sets: Example 4

- Which elements are inside the power sets?
  - P({∅}) =

0: excluded, 1: included

∅	P({∅})
0	{}
1	{∅}

  - P({∅}) = {{}, {∅}} = {∅, {∅}}

Power sets: Example 5

- Which elements are inside the power sets?
  - $P(\{\{\varnothing\}, \varnothing\}) =$

0: excluded, 1: included

$\{\varnothing\}$	$\varnothing$	$P(\{\{\varnothing\}, \varnothing\})$
0	0	$\varnothing$
0	1	$\{\varnothing\}$
1	0	$\{\{\varnothing\}\}$
1	1	$\{\{\varnothing\}, \varnothing\}$

  - $P(\{\{\varnothing\}, \varnothing\}) = \{\varnothing, \{\varnothing\}, \{\{\varnothing\}\}, \{\{\varnothing\}, \varnothing\}\}$

Power sets: Example 6

- Decide whether the following statements true or not!
  - $\{a, b\} \subseteq P(\{a, b\})$
  - $\varnothing \subseteq P(\varnothing)$
  - $\{a, b\} \in P(\{a, b\})$
  - $\varnothing \in P(\varnothing)$
  - $\{3, 5\} \subseteq 2^{\{3, 5\}}$
  - $\{3\} \in 2^{\{3, 5\}}$
  - $\{\varnothing, \{a, b\}\} \subseteq P(\{a, b\})$
  - $\{\{a\}, \{b\}\} \in P(\{a, b\})$
  - $\{\varnothing\} \in 2^{\{3, 5\}}$

Operations with sets: Example 1

- Perform the operations!
  - $(\{1, 2, 5\} \cup \{2, 1\}) \cap \{2, 5, 7\} =$

    $= \{1, 2, 5\} \cap \{2, 5, 7\} =$

    $= \{2, 5\}$

  - $\{0, 3, 5\} \cup (\{3, 7\} \cap \{3, 5, 7\}) =$

    $= \{0, 3, 5\} \cup \{3, 7\} =$

    $= \{0, 3, 5, 7\}$

Operations with sets: Example 2

- Perform the operations!
  - $(\cup\{\{3, 5\}, \{1, 2, 3\}, \{7\}\}) \cap (\cap\{\{3, 1, 5\}, \{3, 2\}, \{7\}\}) =$
    $= \{1, 2, 3, 5, 7\} \cap \{\} =$
    $= \{\}$

  - $(\cup\{\{3\}, \{3, 5\}, \{7\}\}) \cup (\cap\{\{1, 2, 3\}, \{2, 3, 4\}\}) =$
    $= \{3, 5, 7\} \cup \{2, 3\} =$
    $= \{2, 3, 5, 7\}$

Operations with sets: Example 3

- Perform the operations!
  - $(\{1, 2, 5\} - \{5, 7, 9\}) \cup (\{5, 7, 9\} - \{1, 2, 5\}) =$

    $= \{1, 2\} \cup \{7, 9\} =$

    $= \{1, 2, 7, 9\}$

  - $(\{3, 5, 7\} - \{1, 2, 5\}) \cap (\{1, 3, 7, 9\} - \{1, 2, 7\}) =$

    $= \{3, 7\} \cap \{3, 9\} =$

    $= \{3\}$

Operations with sets: Example 4

- Perform the assigned operations!
  - $P(\{2, 3, 5\}) - P(\{3, 5\}) =$

    $= \{\emptyset, \{2\}, \{3\}, \{5\}, \{2, 3\}, \{2, 5\}, \{3, 5\}, \{2, 3, 5\}\} - \{\emptyset, \{3\}, \{5\}, \{3, 5\}\} =$

    $= \{\{2\}, \{2, 3\}, \{2, 5\}, \{2, 3, 5\}\}$

  - $(2^{\{5, 7\}} - 2^{\{5, 9\}}) \cup 2^{\{3, 5\}} =$

    $= (\{\emptyset, \{5\}, \{7\}, \{5, 7\}\} - \{\emptyset, \{5\}, \{9\}, \{5, 9\}\}) \cup \{\emptyset, \{3\}, \{5\}, \{3, 5\}\} =$

    $= \{\{7\}, \{5, 7\}\} \cup \{\emptyset, \{3\}, \{5\}, \{3, 5\}\} =$

    $= \{\emptyset, \{3\}, \{5\}, \{7\}, \{5, 7\}, \{3, 5\}\}$

Operations with sets: Example 5

- Perform the operations!
  - $(\{1, 2\} \cup \{2, 3\}) \cap (\{4, 5, 6\} \cap \{3, 5, 7\}) =$

    $= \{1, 2, 3\} \cap \{5\} =$

    $= \emptyset$

  - $((\{1, 4, 5, 7\} \cup \{2, 4, 8, 3\}) \cap \{1, 2, 3, 4, 5, 6, 7\}) \cap ((\{4, 5, 6\} \cup \{3, 5, 7\}) - \{4, 6, 7\}) =$

    $= (\{1, 2, 3, 4, 5, 7, 8\} \cap \{1, 2, 3, 4, 5, 6, 7\}) \cap (\{3, 4, 5, 6, 7\} - \{4, 6, 7\}) =$

    $= \{1, 2, 3, 4, 5, 7\} \cap \{3, 5\} =$

    $= \{3, 5\}$

Operations with sets: Example 6

- Perform the operations!
  - A, B are sets, $\triangle$ is defined as:
    $A \triangle B = (A \cup B) - (A \cap B)$

  - $(\{1, 2\} \cap \{2, 3\}) \cup (\{4, 5, 6\} \triangle \{3, 5, 7\}) =$
    $= \{2\} \cup \{3, 4, 6, 7\} =$
    $= \{2, 3, 4, 6, 7\}$

  - $(\{1, 2, 3, 4, 5, 6, 7, 8, 9\} \triangle \{1, 2, 3, 4, 5, 6, 9\}) \triangle \{1, 3, 6, 8\} =$
    $= \{7, 8\} \triangle \{1, 3, 6, 8\} =$
    $= \{1, 3, 6, 7\}$

Operations with sets: Example 7

- Perform the operations!
  - A, B are sets, $\triangle$ is defined as:
    $A \triangle B = (A \cup B) - (A \cap B)$
  - $(\{1, 2, 3, 9\} - \{4, 5, 6, 7\}) \triangle \{1, 5\} =$

    $= \{1, 2, 3, 9\} \triangle \{1, 5\} =$

    $= \{2, 3, 5, 9\}$

  - $((\{1, 9\} - \{3, 8, 9\}) \cup (\{4, 6, 8\} \cup \{1, 3, 6, 8\})) \triangle (\{1, 2, 5\} \triangle (\{3, 6, 9\} \cap \{1, 2, 3, 4, 5\})) =$

    $= (\{1\} \cup \{1, 3, 4, 6, 8\}) \triangle (\{1, 2, 5\} \triangle \{3\}) =$

    $= \{1, 3, 4, 6, 8\} \triangle \{1, 2, 3, 5\}$

    $= \{2, 4, 5, 6, 8\}$

Operations with sets: Example 8

- Perform the operations!
  - A: $\{1, 2, 3, 7, 8\}$, B: $\{3, 4, 5\}$, C: $\{5, 6, 7\}$
  - $A \cup B =$

    $\{1, 2, 3, 7, 8\} \cup \{3, 4, 5\} = \{1, 2, 3, 4, 5, 7, 8\}$
  - $B \cap C =$

    $\{3, 4, 5\} \cap \{5, 6, 7\} = \{5\}$
  - $A \triangle (B \cup (C \cup A)) =$

    $\{1, 2, 3, 7, 8\} \triangle (\{3, 4, 5\} \cup (\{5, 6, 7\} \cup \{1, 2, 3, 7, 8\})) =$

    $\{1, 2, 3, 7, 8\} \triangle (\{3, 4, 5\} \cup \{1, 2, 3, 5, 6, 7, 8\}) =$

    $\{1, 2, 3, 7, 8\} \triangle \{1, 2, 3, 4, 5, 6, 7, 8\} = \{4, 5, 6\}$

Cartesian products

- Perform the operations!
  - $\{1, 5, 9\} \times \{b, c, d\} =$

    $\{(1, b), (1, c), (1, d), (5, b), (5, c), (5, d), (9, b), (9, c), (9, d)\}$

  - $\{0\} \times \{1, 2\} \times \{1, 2, 3\} =$

    $\{(0, 1, 1), (0, 1, 2), (0, 1, 3), (0, 2, 1), (0, 2, 2), (0, 2, 3)\}$

Cartesian products

- Perform the operations!
  - $P(\{4, 5\}) \times \{1, 2\} =$

    $\{\emptyset, \{4\}, \{5\}, \{4, 5\}\} \times \{1, 2\} =$

    $\{(\emptyset, 1), (\emptyset, 2), (\{4\}, 1), (\{4\}, 2), (\{5\}, 1), (\{5\}, 2), (\{4, 5\}, 1), (\{4, 5\}, 2)\}$

  - $P(\{2, 4\}) \times P(\{3, 5\}) =$

    $\{\emptyset, \{2\}, \{4\}, \{2, 4\}\} \times \{\emptyset, \{3\}, \{5\}, \{3, 5\}\} =$

    $\{(\emptyset, \emptyset), (\emptyset, \{3\}), (\emptyset, \{5\}), (\emptyset, \{3, 5\}), (\{2\}, \emptyset), (\{2\}, \{3\}), (\{2\}, \{5\}), (\{2\}, \{3, 5\}), (\{4\}, \emptyset), (\{4\}, \{3\}), (\{4\}, \{5\}), (\{4\}, \{3, 5\}), (\{2, 4\}, \emptyset), (\{2, 4\}, \{3\}), (\{2, 4\}, \{5\}), (\{2, 4\}, \{3, 5\})\}$

Cartesian products

- Decide whether the following statements are true or not!
  - $\{a\} \times \{b\} = \{(a, b)\}$
  - $\{a, b\} \subseteq \{a\} \times \{b\}$
  - $\{a, b\} \in \{a\} \times \{b\}$
  - $(a, b) \in \{a\} \times \{b\}$
  - $\{(a, b)\} \subseteq \{a\} \times \{b\}$

Cartesian products

- Decide whether the following statements are true or not!
  - $(a, b) \in (a, b) \times \{a, b\}$

    $= \{((a, b), a), ((a, b), b)\}$
  - $\{a, b\} \in \{b, a\} \times \{b\}$

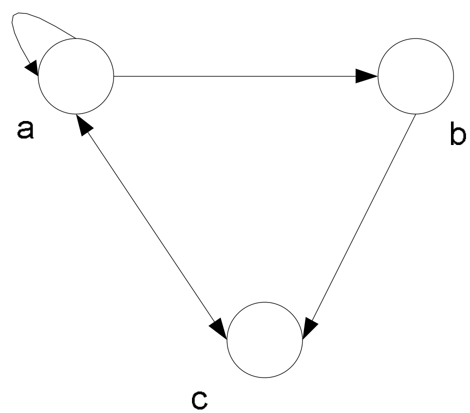
    $= \{(b, b), (a, b)\}$
  - $(a, b) \in \{a, (a, b)\} \times \{(b, a), b\} =$

    $\{(a, (b, a)), (a, b), ((a, b), (b, a)), ((a, b), b)\} =$

    $\{(a, b, a), (a, b), (a, b, b, a), (a, b, b)\}$

Relations: Example 1

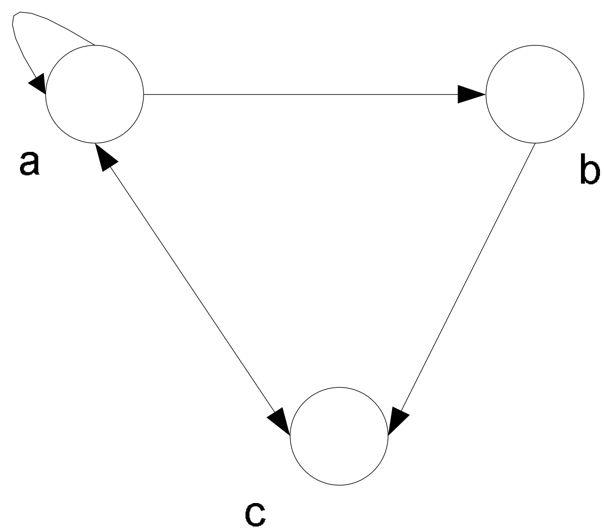
- Is the graph given below is reflexive?
  - $R_1 = \{(a, a), (a, b), (a, c), (b, c), (c, a)\}$
    $R_1 \subseteq \{a, b, c\} \times \{a, b, c\}$



  - answer: no, because there is no loop at each node

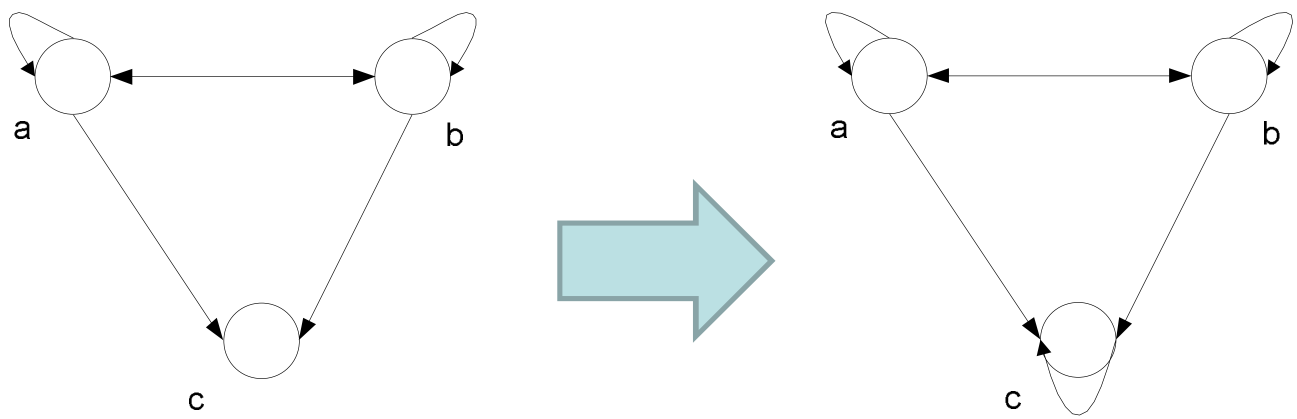
Relations: Example 2

- Decide whether the following statements are true or not!
  - the graph given below is:
    - symmetrical — there are no (b, a), (c, b) edges
    - transitive — (c, b) is missing
    - anti-symmetrical — for (a, c) there is (c, a) edge too



Relations: Example 3

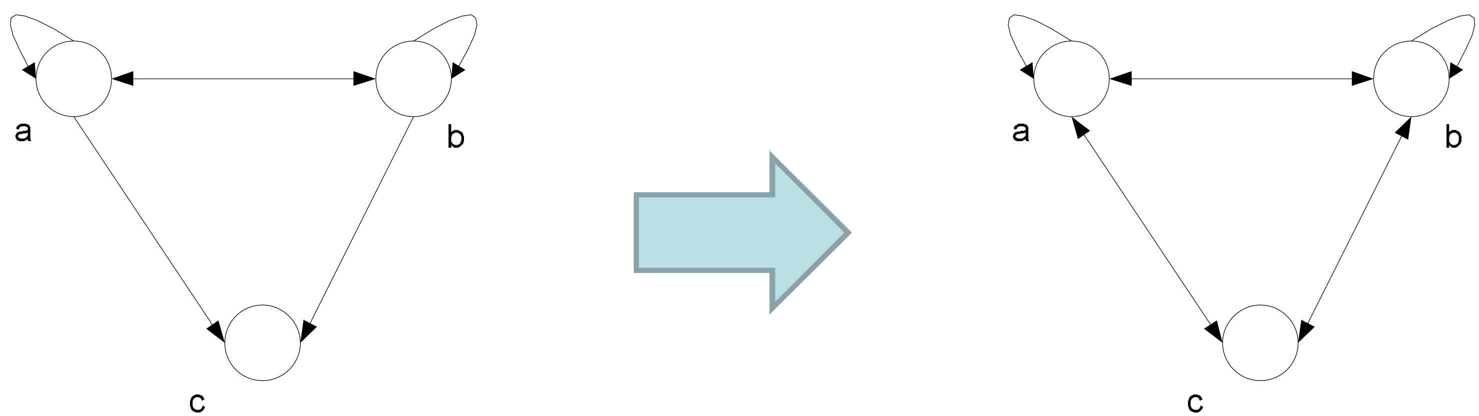
- Calculate the reflexive closure of the graph given below!
  - $R_1 = \{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c)\}$



  - $R_2 = \{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c), (c, c)\}$

Relations: Example 4

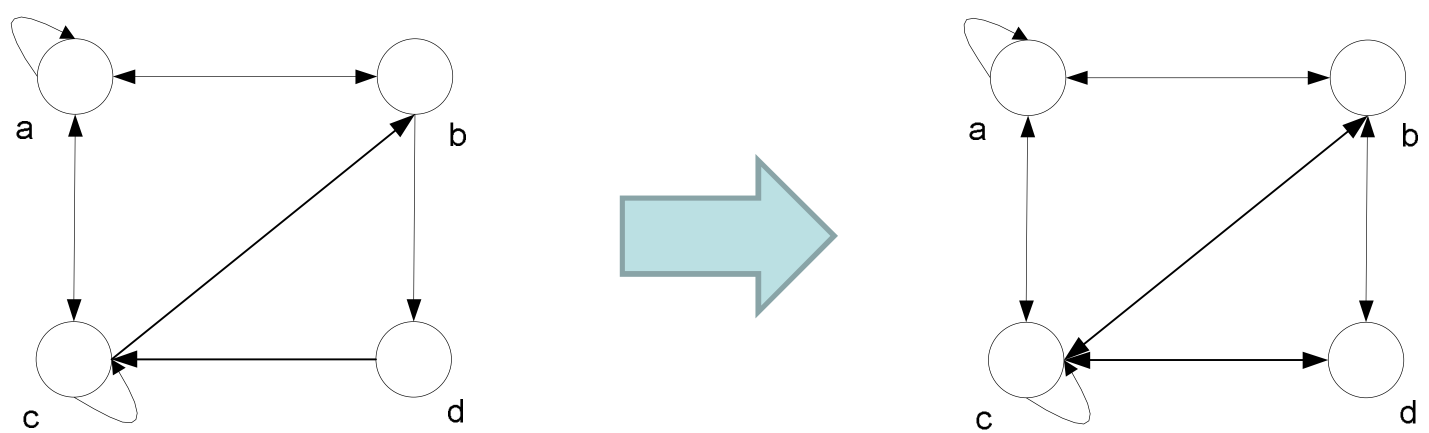
- Calculate the symmetrical closure of the graph given below!
  - $R_1 = \{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c)\}$



  - $R_2 = \{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c), (c, a), (c, b)\}$

Relations: Example 5

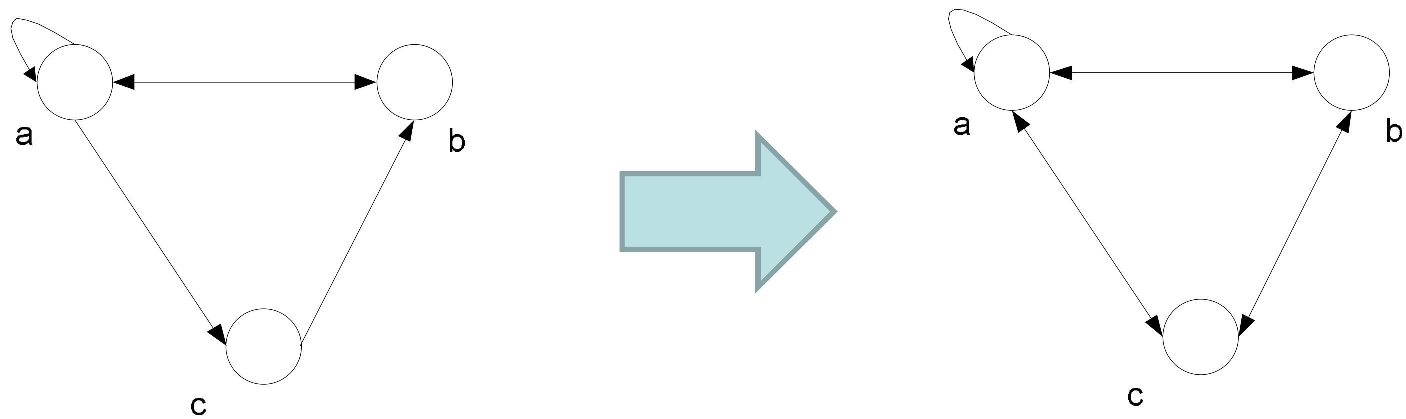
- Calculate the symmetrical closure of the graph given below!
  - $R_1 = \{(a, a), (c, c), (a, b), (b, a), (a, c), (c, a), (d, c), (b, d), (c, b)\}$



  - $R_2 = \{(a, a), (c, c), (a, b), (b, a), (a, c), (c, a), (d, c), (b, d), (c, b), (b, c), (c, d), (d, b)\}$

Relations: Example 6

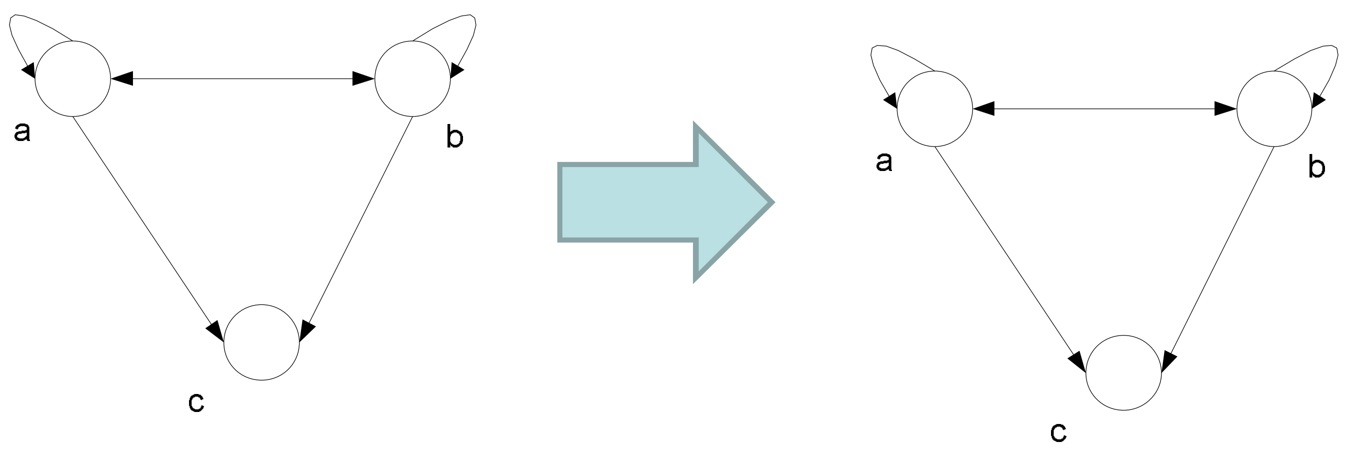
- Calculate the transitive closure of the graph given below!
  - $R_1 = \{(a, a), (a, b), (a, c), (b, a), (c, b)\}$



  - $R_2 = \{(a, a), (a, b), (a, c), (b, a), (c, b), (b, c), (c, a)\}$

Relations: Example 7

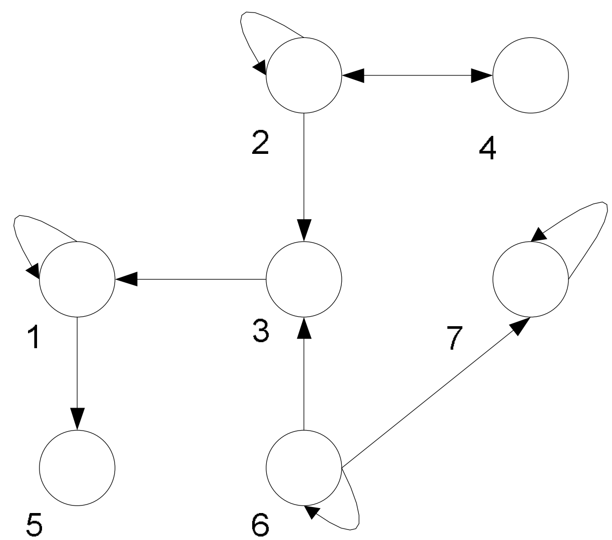
- Calculate the transitive closure of the graph given below!
  - $R_1 = \{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c)\}$



  - $R_2 = \{\{(a, a), (a, b), (b, a), (b, b), (a, c), (b, c)\}$

Relations: Example 8

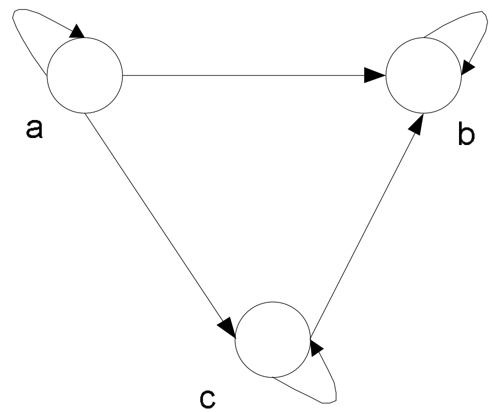
- Describe the properties of this relation!
  - $R_1 = \{(1, 1), (1, 5), (2, 2), (2, 4), (2, 3), (3, 1), (4, 2), (6, 3), (6, 6), (7, 7), (6, 7)\}$



  - not reflexive
  - not symmetric
  - not transitive
  - not anti-symmetric

Relations: Example 9

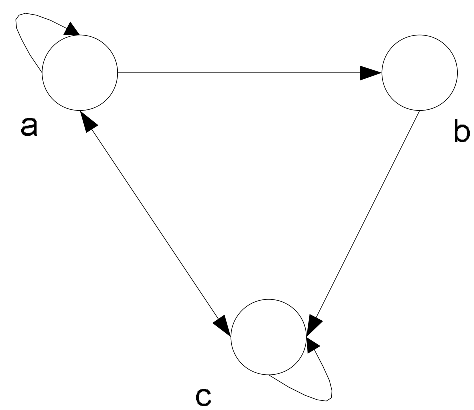
- Describe the properties of this relation!
  - $R_1 = \{(a, b), (b, a), (a, c), (b, b), (c, b), (c, c)\}$



  - reflexive
  - transitive
  - anti-symmetric

Relations: Example 10

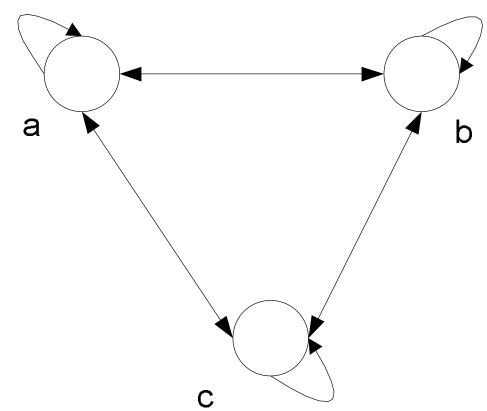
- Describe the properties of this relation!
  - $R_2 = \{(a, a), (a, b), (b, c), (a, c), (c, a), (c, c)\}$



  - not reflexive
  - not symmetric
  - not transitive
  - not anti-symmetric

Relations: Example 11

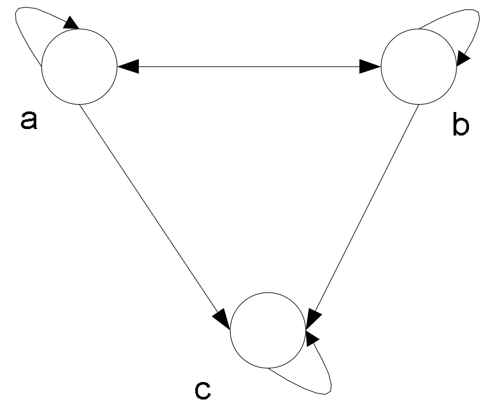
- Describe the properties of the union of previous two relations!
  - $R_1 \cup R_2$ = {(a, a), (a, b), (a, c), (b, a), (b, b), (b, c), (c, a), (c, b), (c, c)}



  - reflexive
  - symmetric
  - transitive

Relations: Example 12

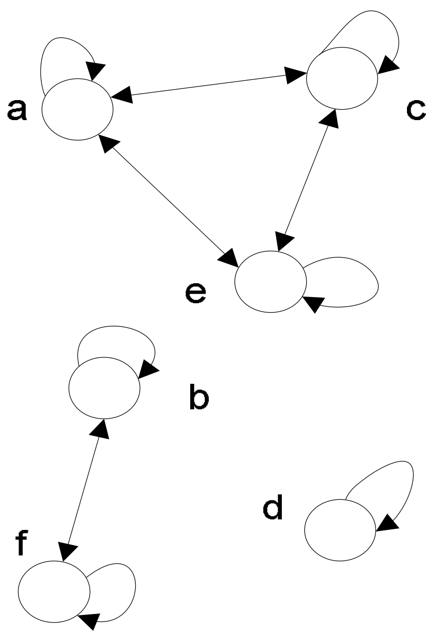
- Decide whether the following statements true or not!
  - R = {(a, a), (b, b), (c, c), (a, b), (b, a), (a, c), (b, c)}



  - the graph is given below:
    - reflexive
    - symmetric
    - anti-symmetric
    - transitive
    - equivalence

Relations: Example 13

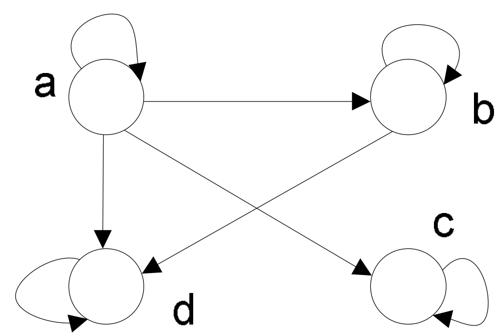
- Decide whether the following statements true or not!
  - R = {(a, a), (a, c), (c, a), (c, c), (c, e), (e, c), (e, e), (e, a), (a, e), (b, b), (b, f), (f, b), (f, f), (d, d)}



- reflexive
- symmetric
- anti-symmetric
- transitive
- equivalence

Relations: Example 14

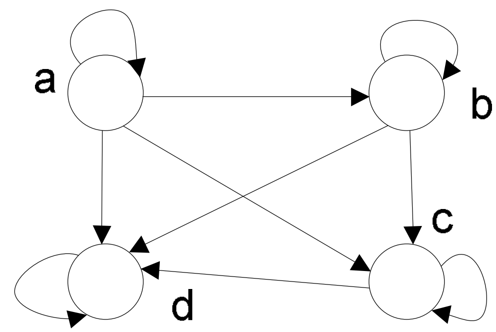
- Decide whether the following statements true or not!
  - R = {(a, a), (b, b), (c, c), (d, d), (a, b), (a, c), (a, d), (b, d)}



  - the graph is given below:
    - reflexive
    - symmetric
    - anti-symmetric
    - transitive
    - equivalence

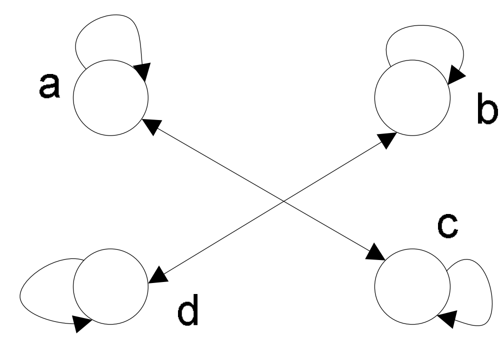
Relations: Example 15

- Decide whether the following statements true or not!
  - R = {(a, a), (b, b), (c, c), (d, d), (a, b), (b, d), (a, d), (a, c), (c, d), (b, c)}
  - the graph is given below:
    - reflexive
    - symmetric
    - anti-symmetric
    - transitive
    - equivalence



Relations: Example 16

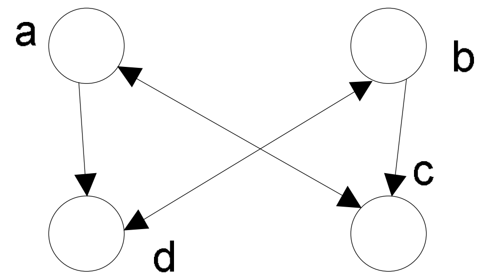
- Decide whether the following statements true or not!
  - R = {(a, a), (b, b), (c, c), (d, d), (a, c), (c, a), (b, d), (d, b)}



  - the graph is given below:
    - reflexive
    - symmetric
    - anti-symmetric
    - transitive
    - equivalence

Relations: Example 17

- Decide whether the following statements true or not!
  - R = {(a, c), (c, a), (b, d), (d, b), (a, d), (b, c)}



  - the graph is given below:
    - reflexive
    - symmetric
    - anti-symmetric
    - transitive
    - equivalence

Element of the Theory of the Computation

Lecture 2

Algorithm complexity: Example 1

- Let $f(n) = 9n^2 + 3n + 2$
  - is it true: $f(n) \in O(n^2)$?
  - $f(n) = 9n^2 + 3n + 2 < 9n^2 + 3n^2 + 2 = 12n^2 + 2$
  - $f(n) \leq c * n^2 + d$
    - $c = 12, d = 2$

- Let $f(n) = 9n^2 + 3n + 2$
  - is it true: $n^2 \in O(f)$?
  - $n^2 \leq c\ (9n^2 + 3n + 2) + d$
    - $c = 1, d = 0$

Algorithm complexity: Example 1

- Let $f(n) = 9n^2 + 3n + 2$
  - is it true: $n^2 \approx f$?
  - $f(n) \in O(n^2)$, $n^2 \in O(f)$

Algorithm complexity: Example 2

- Let $f(n) = 2n^2 + 3n$
  - is it true: $f(n) \in O(n^3)$?
  - $f(n) = 2n^2 + 3n < 2n^3 + 3n^3 = 5n^3$
  - $f(n) \leq c * n^3 + d$
    - $c = 5, d = 0$

- Let $f(n) = 2n^2 + 3n$
  - is it true: $n^3 \in O(f)$?
  - $n^3 \leq c * (2n^2 + 3n) + d$
    - there's no c and d to satisfy the equation for all n

Algorithm complexity: Example 2

- Let $f(n) = 2n^2 + 3n$
  - is it true: $n^3 \approx f$?
  - no, because
    - $n^3 \notin O(f)$
    - $f \in O(n^3)$

Algorithm complexity: Example 3

- Is it true or false?
  - $f(n) = 2^n + n^7$, $g(n) = 5n^7$

    $f(n) \in O(g(n))$
    $g(n) \in O(f(n))$
    $f(n) \approx g(n)$

  - $f(n) = 100$, $g(n) = n$

    $f(n) \in O(g(n))$
    $g(n) \in O(f(n))$
    $f(n) \approx g(n)$

Algorithm complexity: Example 4

- Is it true or false?
  - $f(n) = 5n^4 + n^2 - 700$, $g(n) = 2n^4 + n^3 + 100$

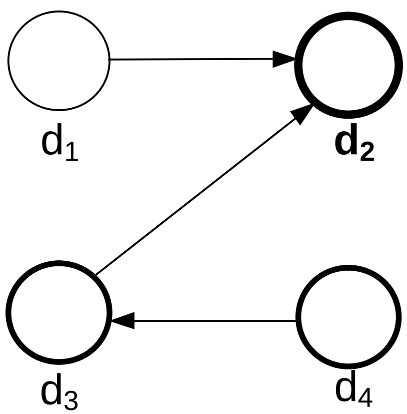
    $f(n) \in O(g(n))$
    $g(n) \in O(f(n))$
    $f(n) \approx g(n)$

  - $f(n) = 20n^{3.1} + 6n^2 + 10n + 6$, $g(n) = 10n^3 + n^{0.1} + 6$

    $f(n) \in O(g(n))$
    $g(n) \in O(f(n))$
    $f(n) \approx g(n)$

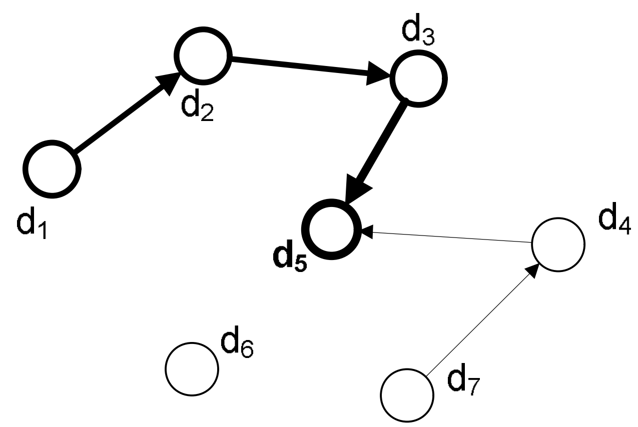
Closure: Example 1

- A binary relation is given on D × D. Give the closure of set A on this relation!
  - $D = \{d_1, d_2, d_3, d_4\}$
  - $A = \{d_4\}$
  - $(d_4, d_3) \in R, d_4 \in A^* \rightarrow d_3 \in A^*$
  - $A^* = \{d_3, d_4\}$
  - $(d_3, d_2) \in R, d_3 \in A^* \rightarrow d_2 \in A^*$
  - $A^* = \{d_2, d_3, d_4\}$



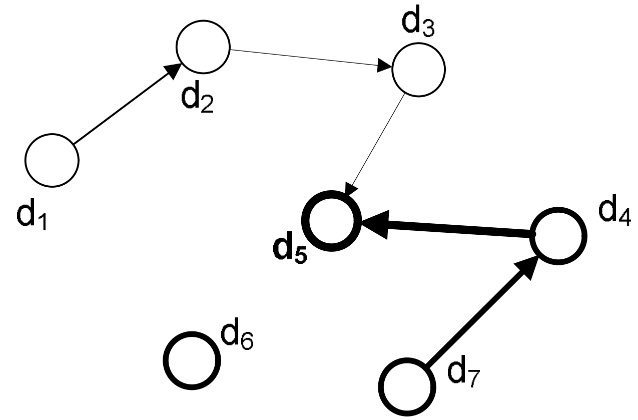
Closure: Example 2

- A binary relation is given on D × D. Give the closure of set A on this relation!
  - $D = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7\}$
  - $A = \{d_1, d_2\}$
  - $A^* = \{d_1, d_2\}$
  - $A^* = \{d_1, d_2, d_3\}$
  - $A^* = \{d_1, d_2, d_3, d_5\}$



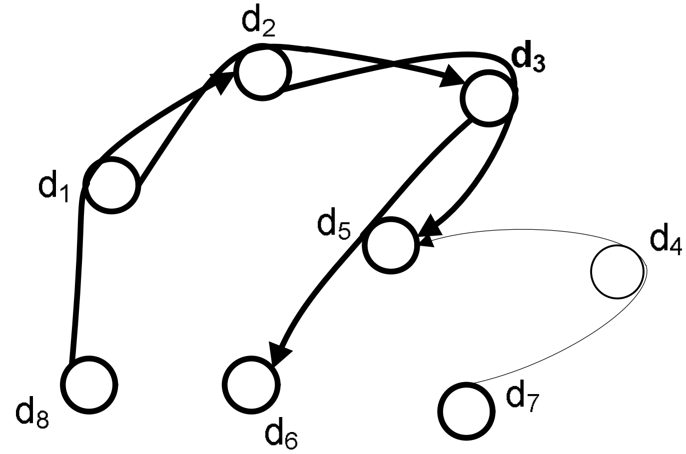
Closure: Example 3

- A binary relation is given on D × D. Give the closure of set A on this relation!
  - $D = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7\}$
  - $A = \{d_6, d_7\}$
  - $A^* = \{d_6, d_7\}$
  - $A^* = \{d_7, d_6, d_4\}$
  - $A^* = \{d_6, d_7, d_4, d_5\}$



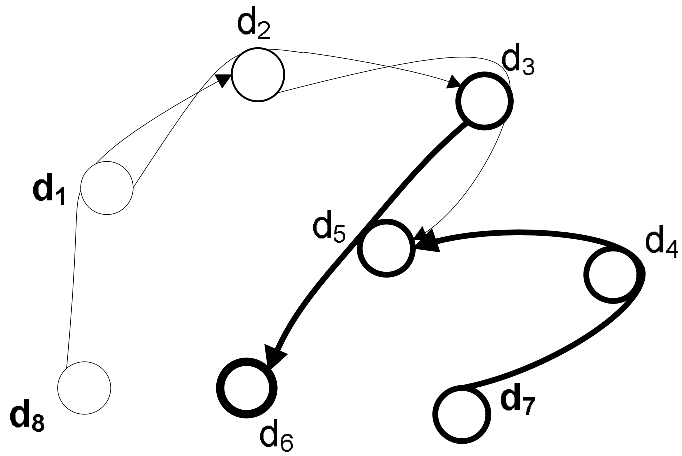
Closure: Example 4

- A ternary relation is given on D × D × D. Give the closure of set A on this relation!
  - $D = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}\}$
  - $A = \{d_8, d_7, d_1\}$
  - $A^* = \{d_8, d_7, d_1\}$
  - $A^* = \{d_8, d_7, d_1, d_2\}$
  - $A^* = \{d_8, d_7, d_1, d_2, d_3\}$
  - $A^* = \{d_8, d_7, d_1, d_2, d_3, d_5\}$
  - $A^* = \{d_8, d_7, d_1, d_2, d_3, d_5, d_6\}$



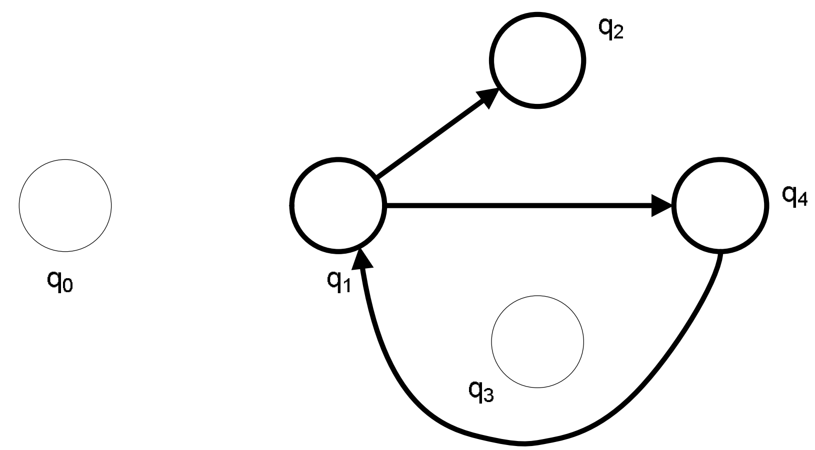
Closure: Example 5

- A ternary relation is given on D × D × D. Give the closure of set A on this relation!
  - $D = \{d_1, d_2, d_3, d_4, d_5, d_6, d_7, d_8, d_9, d_{10}\}$
  - $A = \{d_4, d_7, d_3\}$
  - $A^* = \{d_4, d_7, d_3\}$
  - $A^* = \{d_4, d_7, d_3, d_5\}$
  - $A^* = \{d_4, d_7, d_3, d_5, d_6\}$



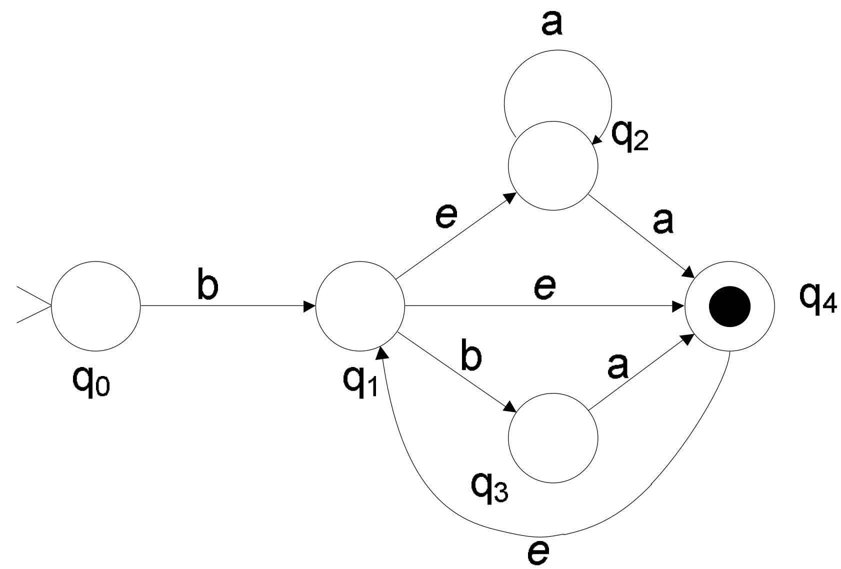
Closure: Example 6

- Give the $E(q_1)$ set of the following NFA!
  - $E(q_1) = \{q_1\}$
  - $E(q_1) = \{q_1, q_2\}$
  - $E(q_1) = \{q_1, q_2, q_4\}$



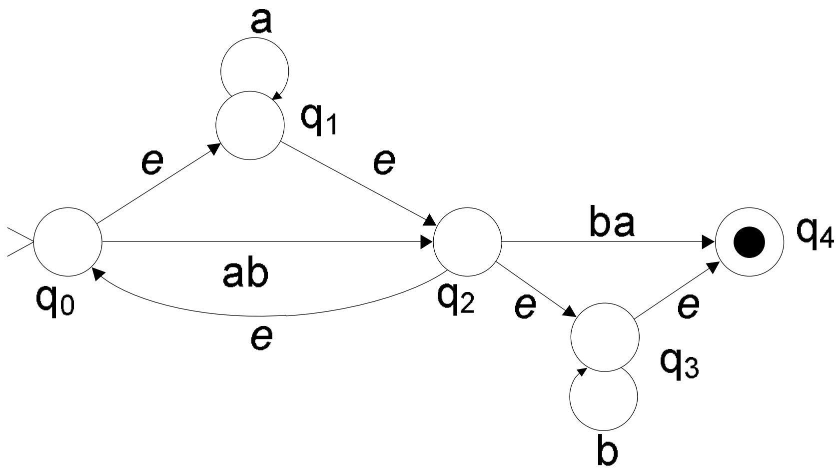
Closure: Example 6

- Give the "E" sets of the states of the following NFA!
  - $E(q_0) = \{q_0\}$
  - $E(q_1) = \{q_1, q_2, q_4\}$
  - $E(q_2) = \{q_2\}$
  - $E(q_3) = \{q_3\}$
  - $E(q_4) = \{q_4, q_1, q_2\}$



Closure: Example 7

- Give the "E" sets of the states of the following NFA!
  - $E(q_0) = \{q_0, q_1, q_2, q_3, q_4\}$
  - $E(q_1) = \{q_0, q_1, q_2, q_3, q_4\}$
  - $E(q_2) = \{q_0, q_1, q_2, q_3, q_4\}$
  - $E(q_3) = \{q_3, q_4\}$
  - $E(q_4) = \{q_4\}$



Element of the Theory of the Computation

Lecture 3

Language: Example 1

- Define the next language: $L(a \cup b)L(c \cup d) =$
- Solution:

$= (L(a) \cup L(b))(L(c) \cup L(d)) =$

$= (\{a\} \cup \{b\})(\{c\} \cup \{d\}) =$

$= \{a, b\}\{c, d\} =$

$= \{ac, ad, bc, bd\}$

Language: Example 2

- Define the next language: $L(abc \cup ab)L(cd \cup d) =$
- Solution:

  $= (L(abc) \cup L(ab))(L(cd) \cup L(d)) =$

  $= (\{abc\} \cup \{ab\})(\{cd\} \cup \{d\}) =$

  $= \{abc, ab\}\{cd, d\} =$

  $= \{abccd, abcd, abd\}$

Language: Example 3

- Define the next language: $L(a \cup \varnothing)L(ab \cup ba) =$
- Solution:

  $= L(a \cup \varnothing)L(ab \cup ba) =$

  $= (L(a) \cup L(\varnothing))(L(ab) \cup L(ba)) =$

  $= (\{a\} \cup \varnothing)(\{ab\} \cup \{ba\}) =$

  $= \{a\}\{ab, ba\} =$

  $= \{aab, aba\}$

Language: Example 4

- Define the next language: $L(a)(L(a \cup b)L(c \cup \varnothing^*))$
- Solution:

$= L(a)((L(a) \cup L(b))(L(c) \cup L(\varnothing^*))) =$

$= \{a\}((\{a\} \cup \{b\})(\{c\} \cup \{e\})) =$

$= \{a\}(\{a, b\}\{c, e\}) =$

$= \{a\}\{ac, a, bc, b\} =$

$= \{aac, aa, abc, ab\}$

Language: Example 5

- Is the statement, $a \in L(a^*b^*a^*b^*)$, true?
- Is the statement, $L(b^*a^*) \cap L(a^*b^*) = L(a^* \cup b^*)$, true?
- Is the statement, $L(a^*b^*) \cap L(c^*d^*) = \emptyset$, true?

  $= \{e\}$
- Is the statement, $abcd \in L((a(cd)^*b^*)^*)$, true?
  - false, because the first iteration of the outermost * can generate "ab" but after that there is a compulsory "a"
  - $abacd \in L((a(cd)^*b^*)^*)$

Language: Example 6

- Which strings are the elements of the following language? $\{w \in \Sigma^* \mid w = uu^Ru, u \in \Sigma^*\}$, $\Sigma = \{a, b\}$
- Solution:
  - {abbaab, baabba, aaaaaa, bbbbbb, ...}

- Which strings are the elements of the following language? $\{w \in \Sigma^* \mid ww = www\}$, $\Sigma = \{a, b\}$
- Solution:
  - $\{e\}$

Regular Expression: Example 1

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^*\}$
- Solution:
  - RE = $(a^*b^*)^*$ or RE = $(a \cup b)^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \text{ is odd}\}$
- Solution:
  - RE = $b^*ab^*(ab^*ab^*)^*$

Regular Expression: Example 2

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \text{ is even}\}$
- Solution:
  – $RE = b^* \cup (b^*ab^*ab^*)^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid w \text{ contains substring "abba"}\}$
- Solution:
  – $RE = (a \cup b)^*abba(a \cup b)^*$

Regular Expression: Example 3

- Give regular expression RE such that
  L(RE) = {w $\in$ {a, b}* | w not contains substring "ab"}
- Solution:
  - RE = b*a*

- Give regular expression RE such that
  L(RE) = {w $\in$ {a, b}* | #a = 0}
- Solution:
  - RE = b*

Regular Expression: Example 4

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a = 2\}$
- Solution:
  - $RE = b^*ab^*ab^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b, c\}^* \mid \#a = \{0, 2, 4\}\}$
- Solution:
  - $RE = (b \cup c)^* \cup ((b \cup c)^*a(b \cup c)^*a(b \cup c)^*) \cup ((b \cup c)^*a(b \cup c)^*a(b \cup c)^*a(b \cup c)^*a(b \cup c)^*)$

Regular Expression: Example 5

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \geq 0\}$
- Solution:
  - $RE = (a \cup b)^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \geq 2\}$
- Solution:
  - $RE = b^*ab^*a(b \cup a)^*$

Regular Expression: Example 6

- Give regular expression RE such that $L(RE) = \{w \in \{a, b, c\}^* \mid \#a = 2\}$
- Solution:
  - RE = $(b \cup c)^*a(b \cup c)^*a(b \cup c)^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \leq 2\}$
- Solution:
  - RE = $b^*(a \cup \varnothing^*)b^*(a \cup \varnothing^*)b^*$

Regular Expression: Example 7

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid 3 \leq \#a \leq 6\}$
- Solution:
  - $RE = b^*ab^*ab^*ab^*(a \cup \emptyset^*)b^*(a \cup \emptyset^*)b^*(a \cup \emptyset^*)b^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 4 = 0\}$
- Solution:
  - $RE = b^* \cup (b^*ab^*ab^*ab^*ab^*)^*$

Regular Expression: Example 8

- Give regular expression RE such that
  $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 6 = 2\}$
- Solution:
  - RE = (b*ab*ab*ab*ab*ab*ab*)*b*ab*ab*

- Give regular expression RE such that
  $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 4 = 2 \text{ or } 3\}$
- Solution:
  - RE = (b*ab*ab*ab*ab*)*b*ab*ab*(a ∪ Ø*)b*

Regular Expression: Example 9

- Give regular expression RE such that $L(RE) = \{w \in \{a, b, c\}^* \mid \#a + \#c = 4\}$
- Solution:
  - RE $= b^*(a \cup c)b^*(a \cup c)b^*(a \cup c)b^*(a \cup c)b^*$

- Give regular expression RE such that $L(RE) = \{w \in \{a, b, c\}^* \mid (\#a + \#c) \bmod 4 = 2\}$
- Solution:
  - RE $= (b^*(a \cup c)b^*(a \cup c)b^*(a \cup c)b^*(a \cup c)b^*)^* \; b^*(a \cup c)b^*(a \cup c)b^*$

Regular Expression: Example 10

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a = 2+3k, k \geq 0\}$
- Solution:
  - $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 3 = 2\}$
  - RE = (b*ab*ab*ab*)*(b*ab*ab*)

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \text{ is even or } \#a \bmod 3 = 0\}$
- Solution:
  - RE =(b*ab*ab*ab*)* ∪ (b*ab*ab*)* ∪ b*

Regular Expression: Example 11

- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \text{ is even and } \#a \bmod 3 = 0\}$
- Solution:
  - $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 6 = 0\}$
  - $RE = (b^*ab^*ab^*ab^*ab^*ab^*ab^*)^* \cup b^*$

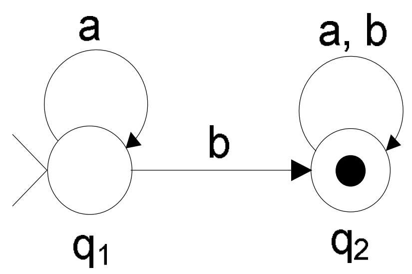
- Give regular expression RE such that $L(RE) = \{w \in \{a, b\}^* \mid \#a \text{ is odd and } \#a \bmod 3 = 0\}$
- Solution:
  - $L(RE) = \{w \in \{a, b\}^* \mid \#a \bmod 6 = 3\}$
  - $RE = b^*ab^*ab^*ab^*(b^*ab^*ab^*ab^*ab^*ab^*ab^*)^*$

Element of the Theory of the Computation

Lecture 4

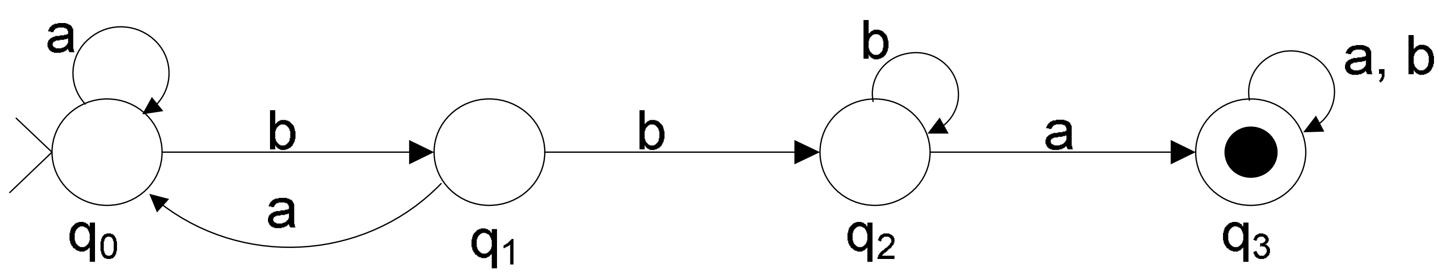
DFA: Example 1

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid w \text{ contains } b\}$
- States:
  - q_1: DFA haven't read 'b' yet
  - q_2: final state – DFA have already read 'b'



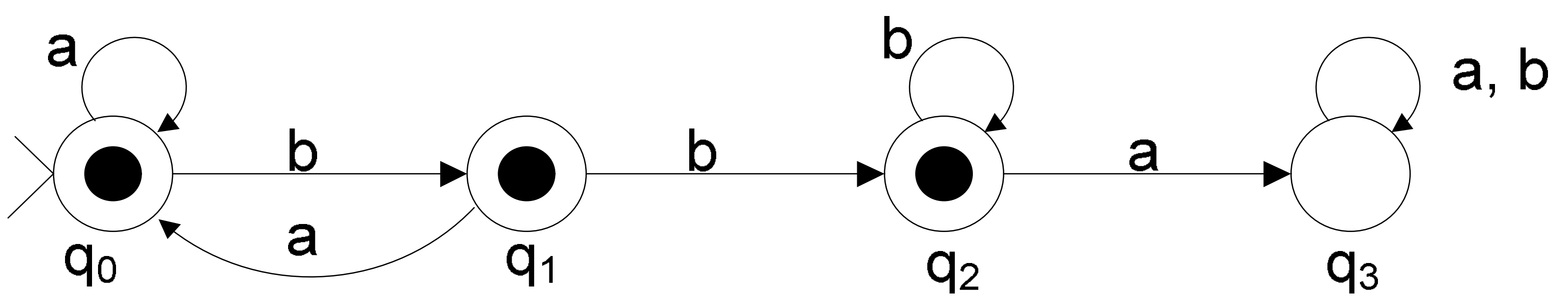
DFA: Example 2

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid w \text{ contains 'bba'}\}$
- States:
  - q_0: 0 symbol is read from 'bba'
  - q_1: 1 symbol is read from 'bba'
  - q_2: 2 symbols are read from 'bba'
  - q_3: 3 symbols are read from 'bba', final state



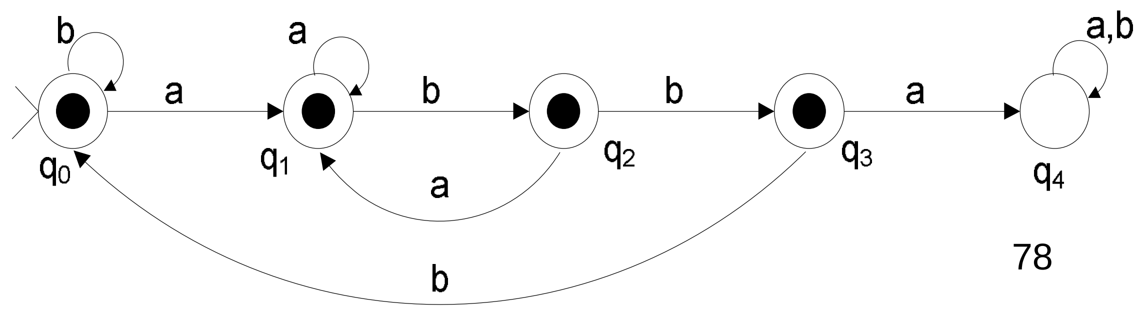
DFA: Example 3

- Define DFA M such that
  - L(M) = {w $\in$ {a, b}* | w does not contain "bba"}
- States:
  - q_0: 0 symbol is read from "bba"
  - q_1: 1 symbol is read from "bba"
  - q_2: 2 symbols are read from "bba"
  - q_3: "bba" is read



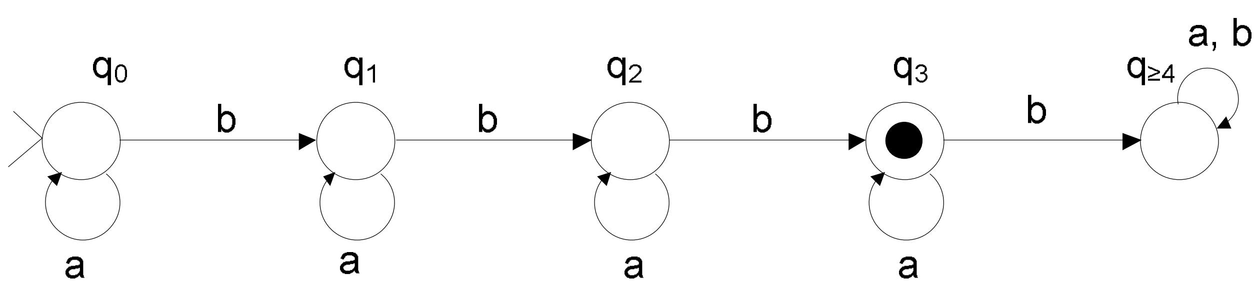
DFA: Example 4

- Define DFA M such that
  - L(M) = {w $\in$ {a, b}* | w does not contain 'abba'}
- States:
  - q_0: 0 symbol is read from 'abba'
  - q_1: 1 symbol is read form 'abba'
  - q_2: 2 symbols are read from 'abba'
  - q_3: 3 symbols are read from 'abba'
  - q_4: 4 symbols are read from 'abba'



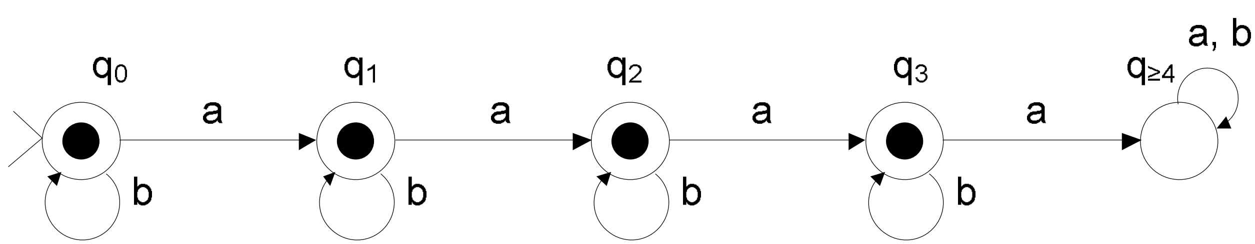
DFA: Example 5

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#b = 3\}$
- States:
  - q_0: $\#b = 0$
  - q_1: $\#b = 1$
  - q_2: $\#b = 2$
  - q_3: $\#b = 3$
  - $q_{\geq 4}$: $\#b \geq 4$



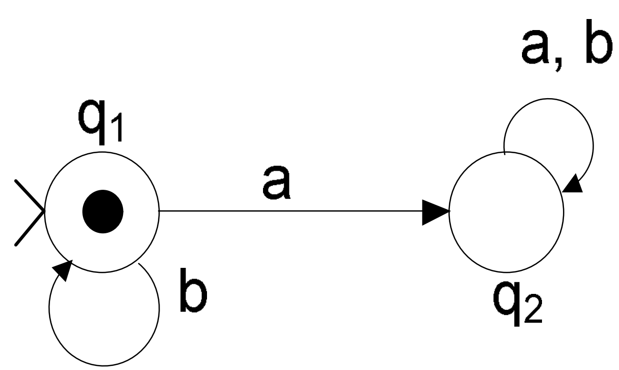
DFA: Example 6

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a < 4\}$
- States:
  - q_0: $\#a = 0$
  - q_1: $\#a = 1$
  - q_2: $\#a = 2$
  - q_3: $\#a = 3$
  - $q_{\geq 4}$: $\#a \geq 4$



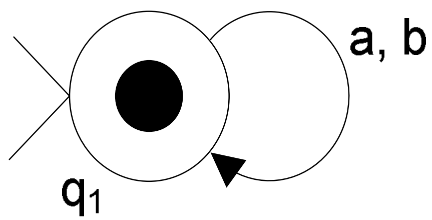
DFA: Example 7

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a = 0\}$
- States:
  - q_1: 0 'a' is read
  - q_2: 1 or more 'a' is read



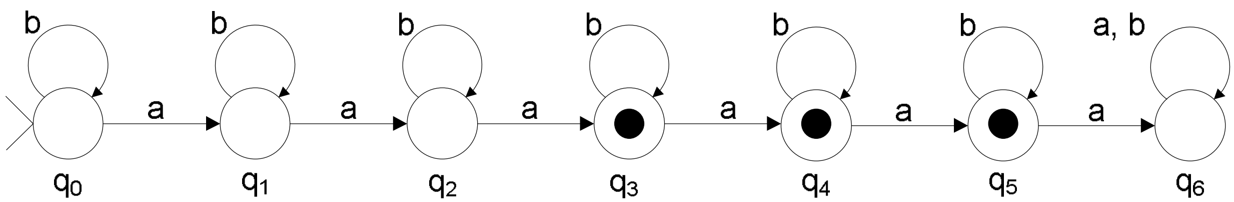
DFA: Example 8

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a \geq 0\}$
- State:
  - q_1: initial and final state



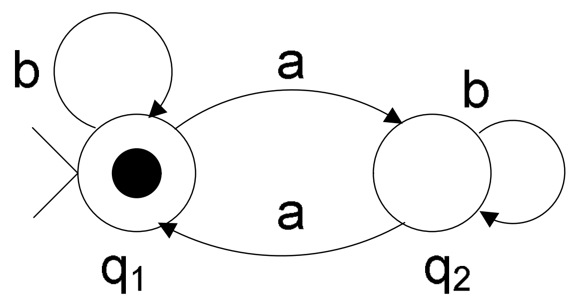
DFA: Example 9

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid 2 < \#a \leq 5\}$
- States:
  - q_0: #a = 0
  - q_1: #a = 1
  - q_2: #a = 2
  - q_3: #a = 3
  - q_4: #a = 4
  - q_5: #a = 5
  - q_6: #a ≥ 6



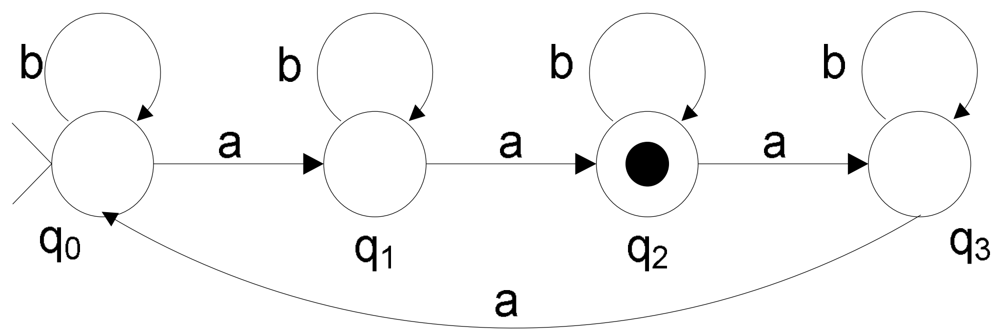
DFA: Example 10

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a \bmod 2 = 0\}$
- States:
  - q_1: #a is even
  - q_2: #a is odd



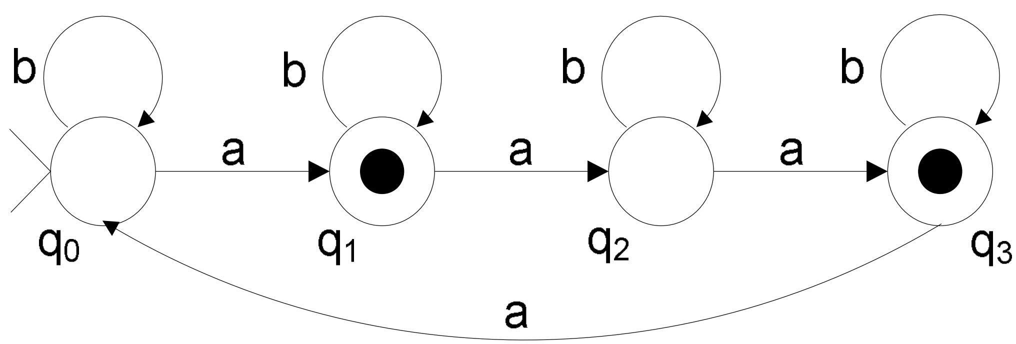
DFA: Example 11

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a \bmod 4 = 2\}$
- States:
  - q_0: #a mod 4 = 0
  - q_1: #a mod 4 = 1
  - q_2: #a mod 4 = 2
  - q_3: #a mod 4 = 3



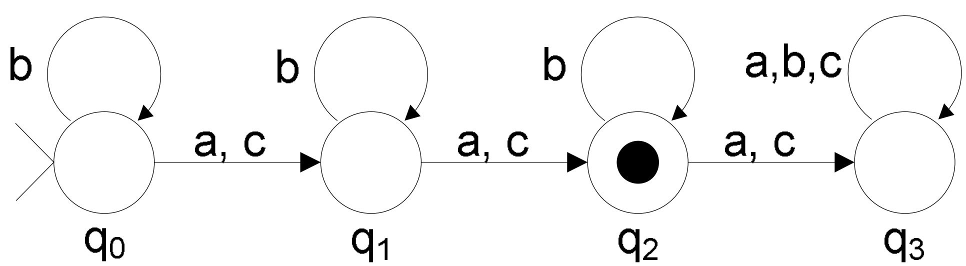
DFA: Example 12

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid \#a \bmod 4 = 1 \text{ or } 3\}$
- States:
  - q_0: #a mod 4 = 0
  - q_1: #a mod 4 = 1
  - q_2: #a mod 4 = 2
  - q_3: #a mod 4 = 3



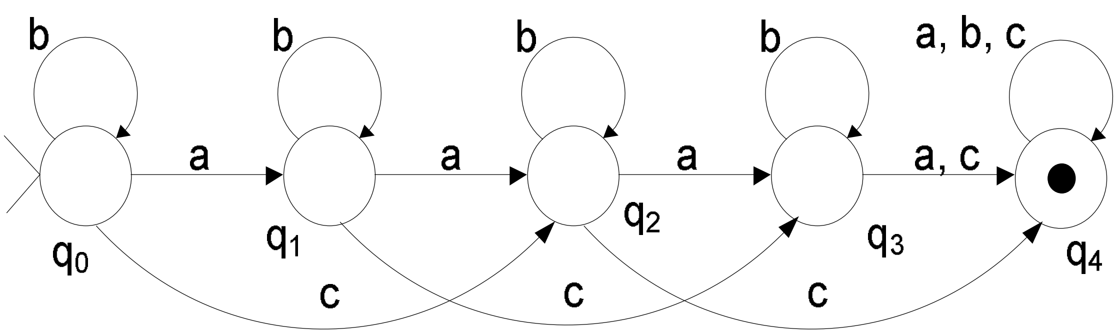
DFA: Example 13

- Define DFA M such that
  - $L(M) = \{w \in \{a, b, c\}^* \mid \#a + \#c = 2\}$
- States:
  - q_0: $\#a + \#c = 0$
  - q_1: $\#a + \#c = 1$
  - q_2: $\#a + \#c = 2$
  - q_3: $\#a + \#c \geq 3$



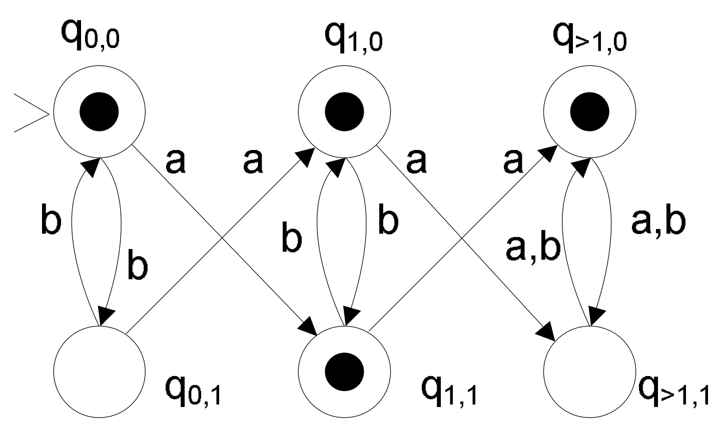
DFA: Example 14

- Define DFA M such that
  - $L(M) = \{w \in \{a, b, c\}^* \mid \#a + 2\#c \geq 4\}$
- States:
  - q_0: $\#a + 2\#c = 0$
  - q_1: $\#a + 2\#c = 1$
  - q_2: $\#a + 2\#c = 2$
  - q_3: $\#a + 2\#c = 3$
  - q_4: $\#a + 2\#c \geq 4$



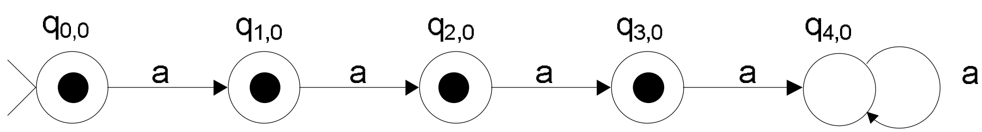
DFA: Example 15

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid (\#a = 1) \text{ or } ((\#a + \#b) \bmod 2 = 0)\}$
- States: two independent conditions
  - $q_{0,j}$: $\#a = 0$
  - $q_{1,j}$: $\#a = 1$
  - $q_{>1,j}$: $\#a > 1$
  - $q_{i,0}$: $(\#a + \#b) \bmod 2 = 0$
  - $q_{i,1}$: $(\#a + \#b) \bmod 2 = 1$



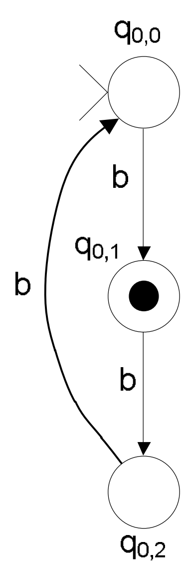
DFA: Example 16

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid (\#a \leq 3) \;\&\; (\#b \bmod 3 = 1)\}$
- States: $q_{i,j}$: #a = i and #b mod 3 = j



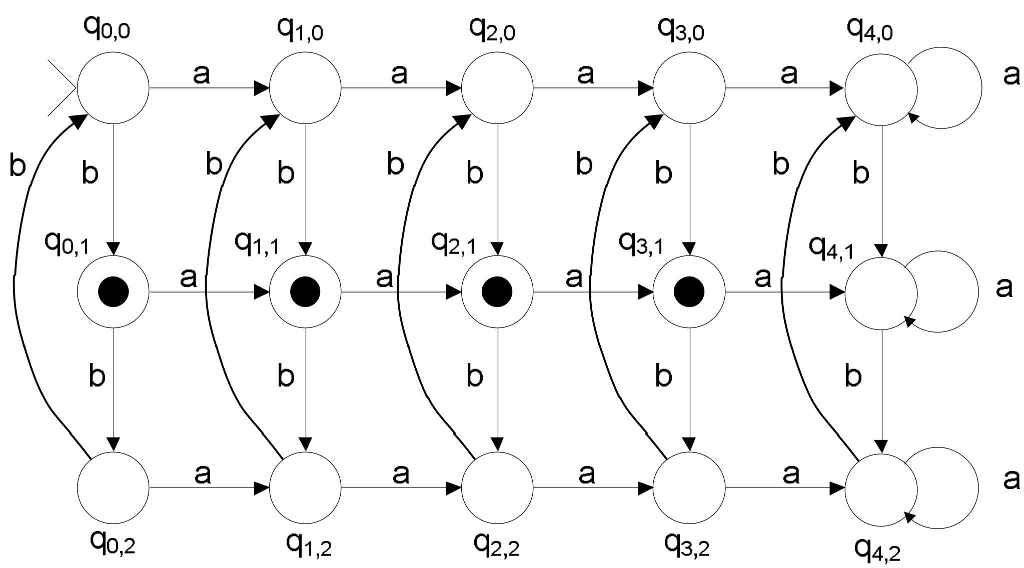
DFA: Example 16

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid (\#a \leq 3) \;\&\; (\#b \bmod 3 = 1)\}$
- States: $q_{i,j}$: $\#a = i$ and $\#b \bmod 3 = j$



DFA: Example 16

- Define DFA M such that
  - $L(M) = \{w \in \{a, b\}^* \mid (\#a \leq 3) \ \& \ (\#b \bmod 3 = 1)\}$
- States: $q_{i,j}$: #a = i and #b mod 3 = j



DFA: Examples

- Define DFA M such that
  - $L = \{w \in \{a, b\}^* \mid w \text{ contains "aabbaabb"}\}$
  - $L = \{w \in \{a, b\}^* \mid \#b = 3 \text{ or } 6\}$
  - $L = \{w \in \{a, b\}^* \mid 3 < a\# \leq 6\}$
  - $L = \{w \in \{a, b\}^* \mid \#a \bmod 5 = 3\}$
  - $L = \{w \in \{a, b\}^* \mid \#b \bmod 3 = 1 \text{ or } 2\}$
  - $L = \{w \in \{a, b, c\}^* \mid \#a + 3\#c = 1\}$
  - $L = \{w \in \{a, b, c\}^* \mid 2\#a + \#b + 3\#c < 7\}$
  - $L(M) = \{w \in \{a, b\}^* \mid (\#b \leq 4) \ \& \ (\#b \bmod 5 = 2 \text{ or } 3)\}$

DFA

- Give the transition diagram of the DFA!
  - $M = (K, \Sigma, \delta, s, F)$
  - $K = \{q_0, q_1\}$
  - $\Sigma = \{a, b\}$
  - $s = q_0$
  - $F = \{q_1\}$
- $L(M) = \{w: \#a \text{ is odd}\}$

q	**σ**	**δ (q, σ)**
q_0	a	q_1
q_0	b	q_0
q_1	a	q_0
q_1	b	q_1

DFA

- Process bbaba with the DFA!

  $(q_0, \text{bbaba}) \vdash_M (q_0, \text{baba})$

  $\vdash_M (q_0, \text{aba})$

  $\vdash_M (q_1, \text{ba})$

  $\vdash_M (q_1, \text{a})$

  $\vdash_M (q_0, e)$

- bbaba is rejected, because q_0 is not a final state

DFA

- Give the transition diagram of the DFA!
  - $M = (K, \Sigma, \delta, s, F)$
  - $K = \{q_0, q_1, q_2\}$
  - $\Sigma = \{a, b\}$
  - $s = q_0$
  - $F = \{q_1\}$
- $L(M) = \{w: \#a \geq 2\}$

q	**σ**	**δ (q, σ)**
q_0	a	q_1
q_0	b	q_0
q_1	a	q_2
q_1	b	q_1
q_2	a	q_2
q_2	b	q_2

DFA

- Process abbbaba with the DFA!

  $(q_0, \text{abbbaba}) \vdash_M (q_1, \text{bbbaba})$

  $\vdash_M (q_1, \text{bbaba})$

  $\vdash_M (q_1, \text{baba})$

  $\vdash_M (q_1, \text{aba})$

  $\vdash_M (q_2, \text{ba})$

  $\vdash_M (q_2, \text{a})$

  $\vdash_M (q_2, e)$

- abbbaba is accepted

Element of the Theory of the Computation

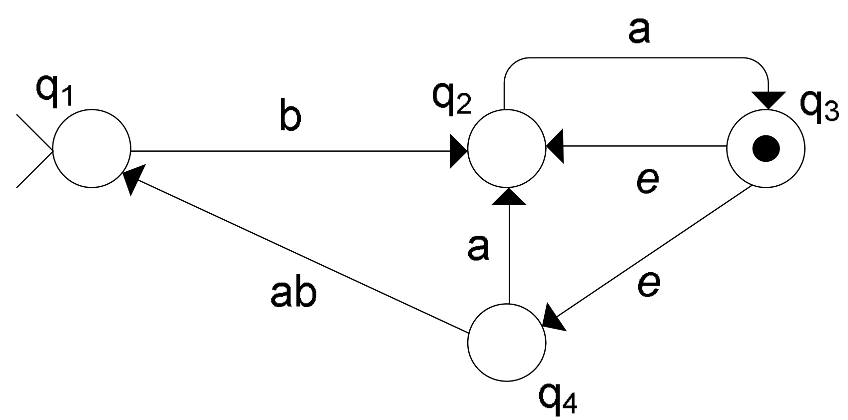
Lecture 5

NFA → DFA

- Steps of NFA to DFA conversion
  - eliminating string transitions by introducing new states
  - definition of e-sets
  - definition of new initial state
  - definition of transition function
  - definition of finite states

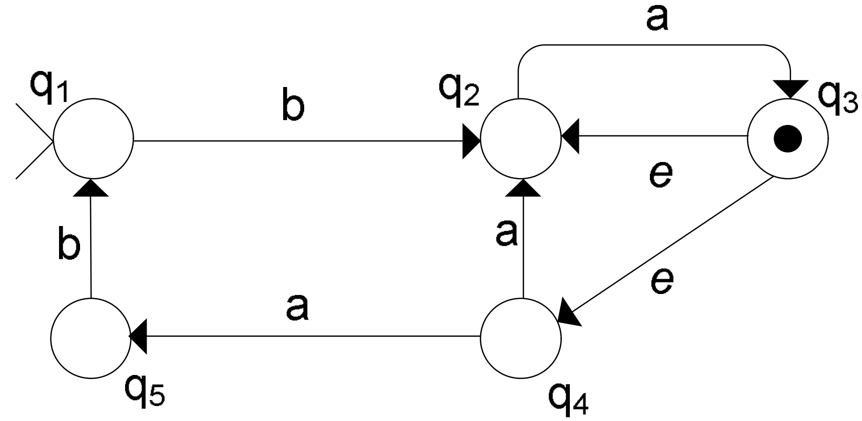
NFA → DFA: Example 1

- Give DFA M' such that L(M') = L(M) !
  - M is given below



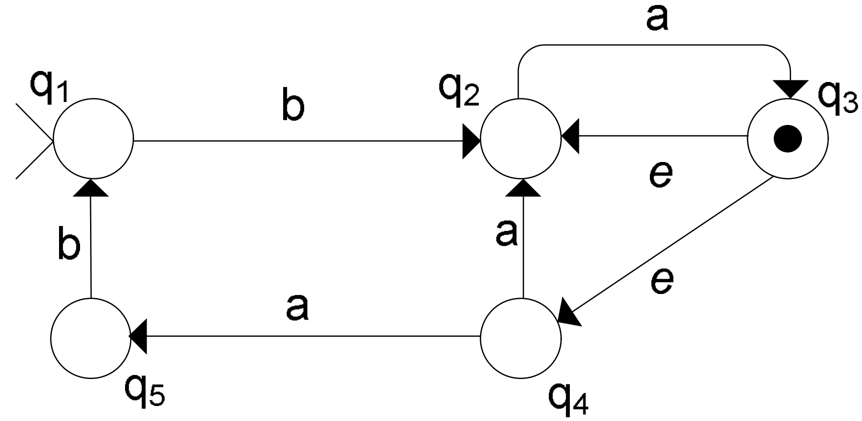
NFA → DFA: Example 1

- Eliminating string transitions by introducing new states



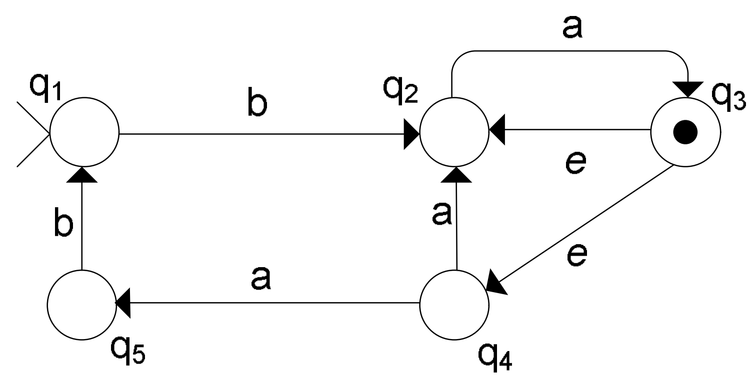
NFA → DFA: Example 1

- Definition of E-sets:
  - $E(q_1) = \{q_1\}$
  - $E(q_2) = \{q_2\}$
  - $E(q_3) = \{q_2, q_3, q_4\}$
  - $E(q_4) = \{q_4\}$
  - $E(q_5) = \{q_5\}$
- Definition of new initial state
  - $s_3 = E(q_1) = \{q_1\} = Q_1$



NFA → DFA: Example 1

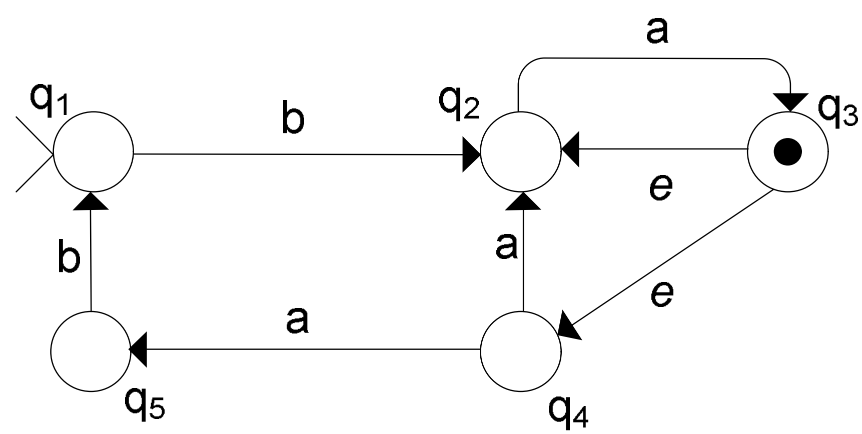
- Definition of transition function:
  - $\delta(\{q_1\}, a) = \emptyset = Q_2$
  - $\delta(\{q_1\}, b) = E(q_2) = \{q_2\} = Q_3$
  - $\delta(\emptyset, a) = \emptyset = Q_2$
  - $\delta(\emptyset, b) = \emptyset = Q_2$
  - $\delta(\{q_2\}, a) = E(q_3) = \{q_2, q_3, q_4\} = Q_4$
  - $\delta(\{q_2\}, b) = \emptyset = Q_2$
  - $\delta(\{q_2, q_3, q_4\}, a) = E(q_2) \cup E(q_3) \cup E(q_5) = \{q_2, q_3, q_4, q_5\} = Q_5$
  - $\delta(\{q_2, q_3, q_4\}, b) = \emptyset = Q_2$



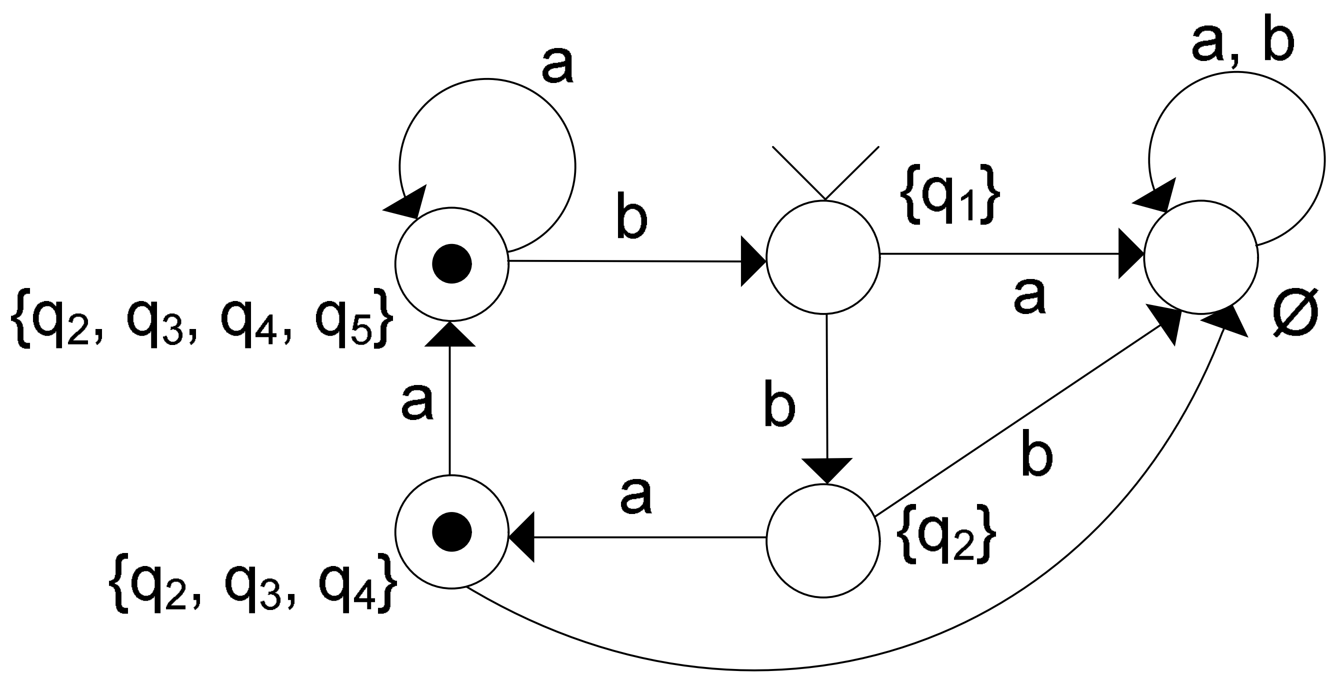
NFA → DFA: Example 1

- $\delta(\{q_2, q_3, q_4, q_5\}, a) = E(q_2) \cup E(q_3) \cup E(q_5) = \{q_2, q_3, q_4, q_5\} = Q_5$
- $\delta(\{q_2, q_3, q_4, q_5\}, b) = E(q_1) = \{q_1\} = Q_1$

- Definition of final states
  - $F_3 = \{\{q_2, q_3, q_4\}, \{q_2, q_3, q_4, q_5\}\}$

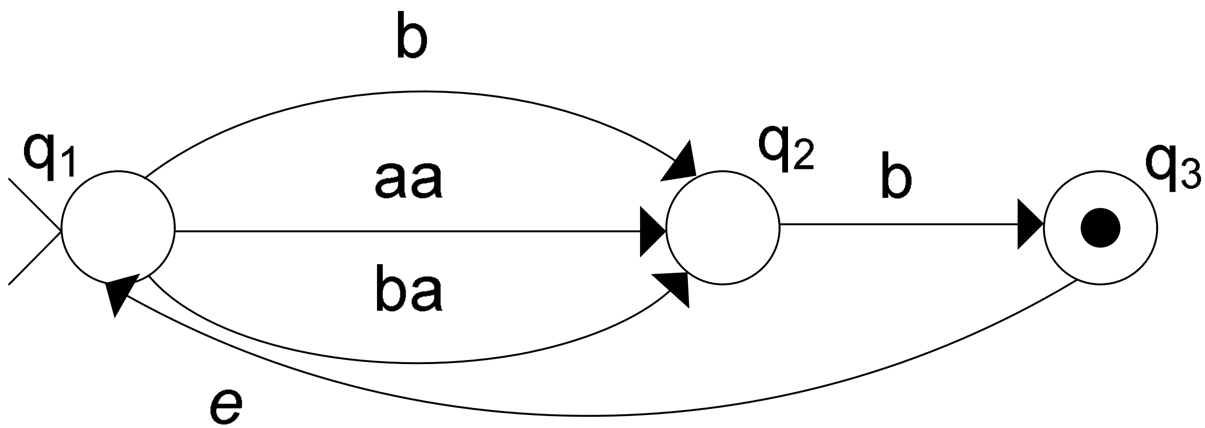


NFA → DFA: Example 1



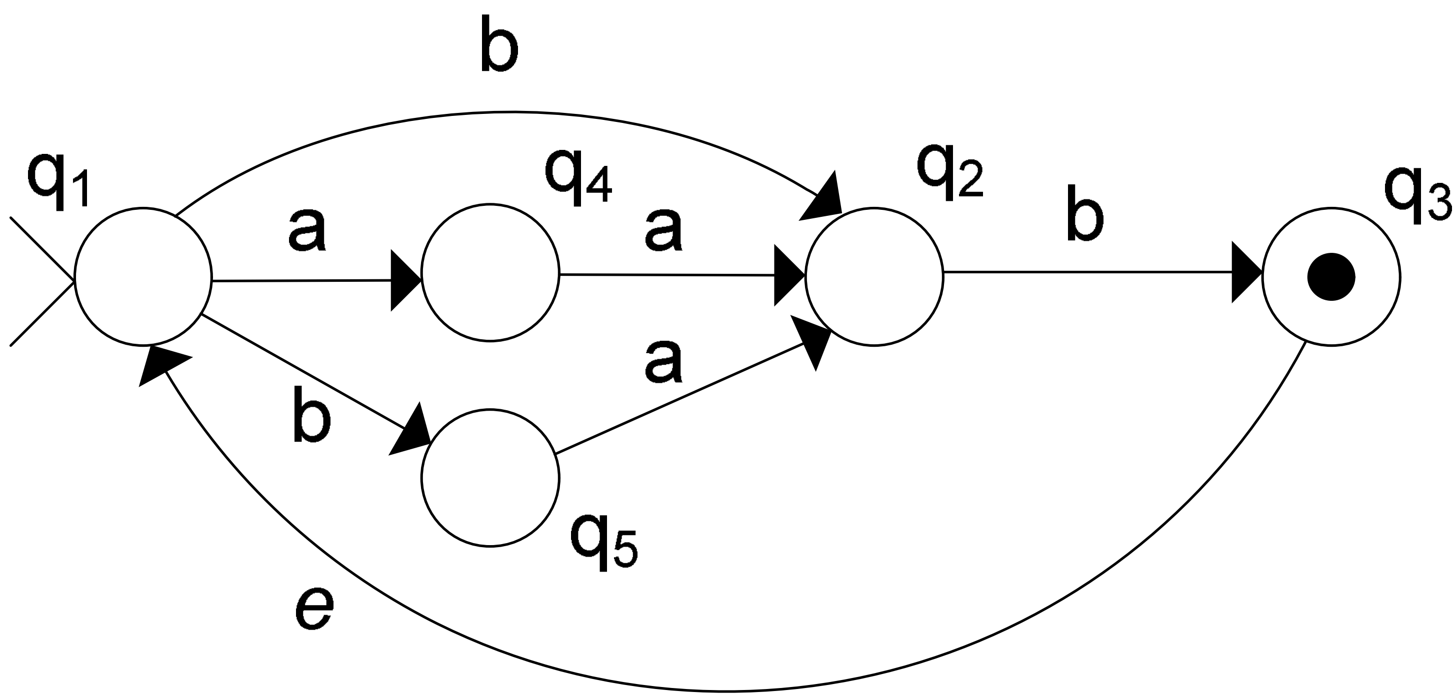
NFA → DFA: Example 2

- Give DFA M' such that L(M') = L(M) !
  - M is given below



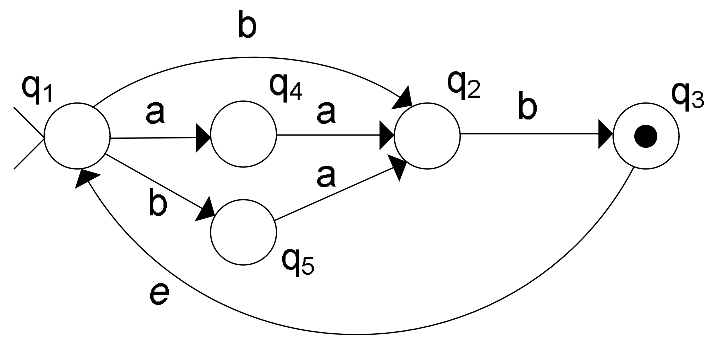
NFA → DFA: Example 2

- Eliminating string transitions by introducing new states



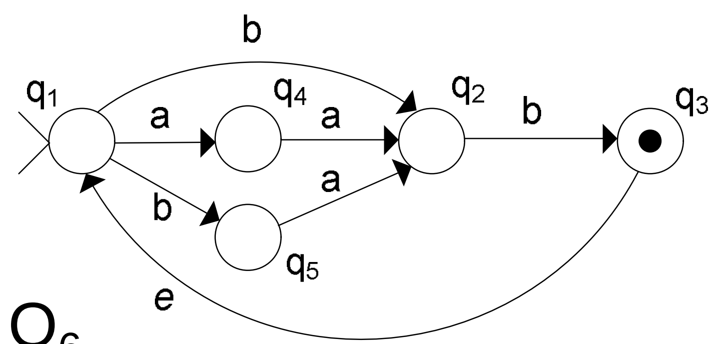
NFA → DFA: Example 2

- Definition of E-sets:
  - $E(q_1) = \{q_1\}$
  - $E(q_2) = \{q_2\}$
  - $E(q_3) = \{q_1, q_3\}$
  - $E(q_4) = \{q_4\}$
  - $E(q_5) = \{q_5\}$
- Definition of new initial state
  - $s_3 = E(q_1) = \{q_1\} = Q_1$



NFA → DFA: Example 2

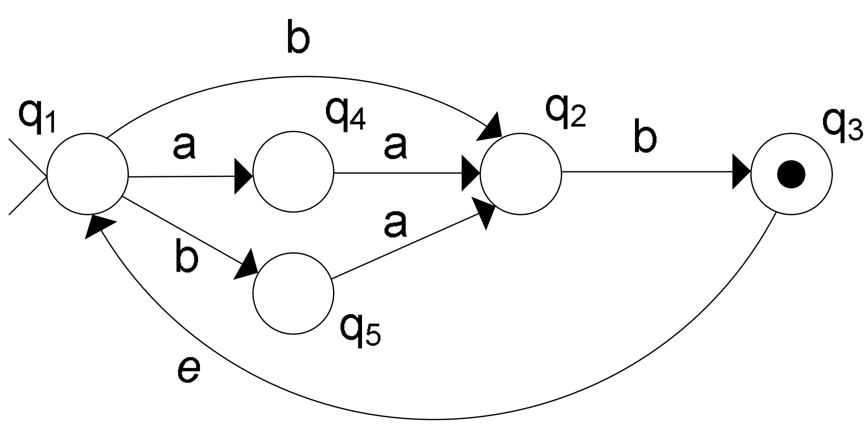
- Definition of transition function:
  - $\delta(\{q_1\}, a) = E(q_4) = \{q_4\} = Q_2$
  - $\delta(\{q_1\}, b) = E(q_2) \cup E(q_5) = \{q_2, q_5\} = Q_3$
  - $\delta(\{q_4\}, a) = E(q_2) = \{q_2\} = Q_4$
  - $\delta(\{q_4\}, b) = \emptyset = Q_5$
  - $\delta(\{q_2, q_5\}, a) = E(q_2) = \{q_2\} = Q_4$
  - $\delta(\{q_2, q_5\}, b) = E(q_3) = \{q_1, q_3\} = Q_6$
  - $\delta(\{q_2\}, a) = \emptyset = Q_5$
  - $\delta(\{q_2\}, b) = E(q_3) = \{q_1, q_3\} = Q_6$



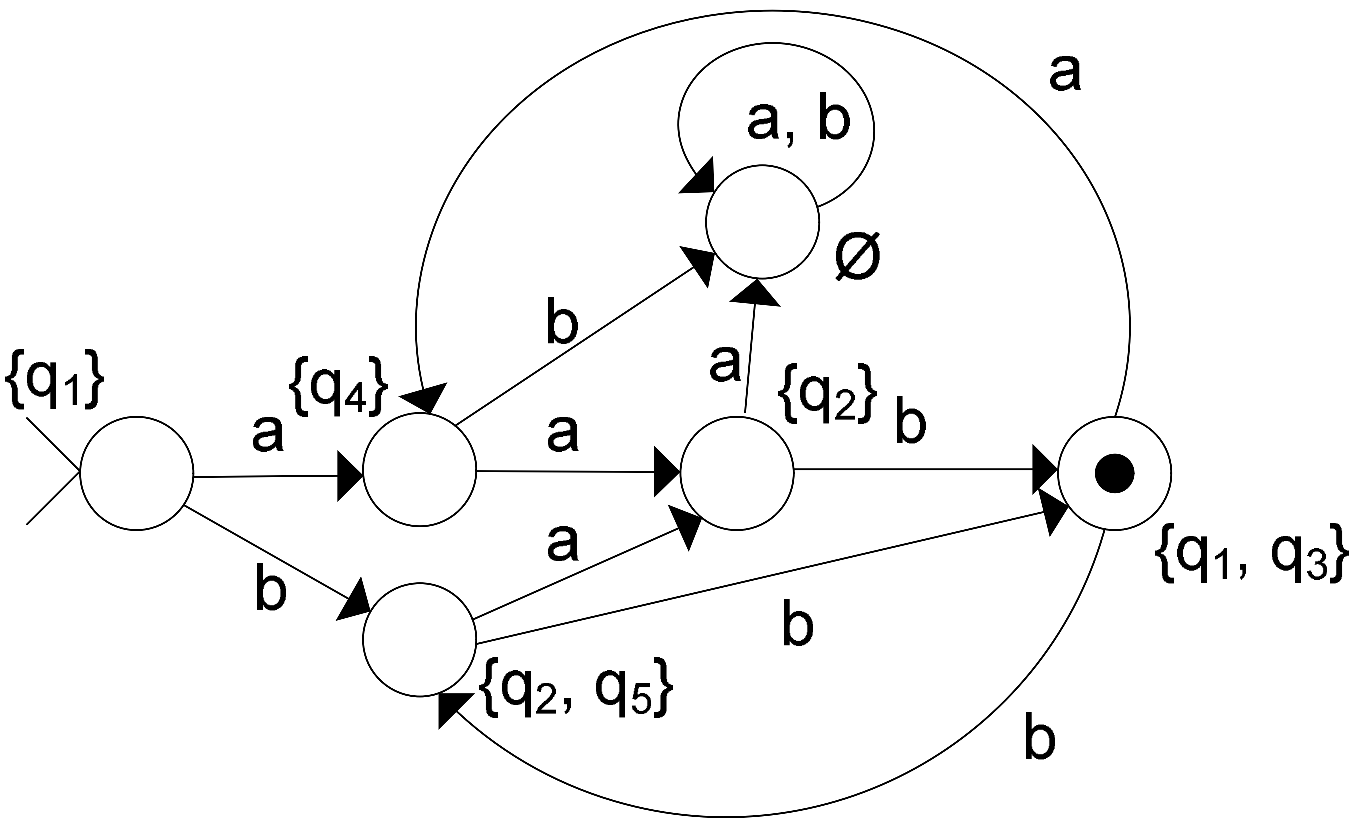
NFA → DFA: Example 2

- $\delta(\varnothing, a) = \varnothing = Q_5$
- $\delta(\varnothing, b) = \varnothing = Q_5$
- $\delta(\{q_1, q_3\}, a) = E(q_4) = \{q_4\} = Q_2$
- $\delta(\{q_1, q_3\}, b) = E(q_2) \cup E(q_5) = \{q_2, q_5\} = Q_3$

• Definition of final states
  - $F_3 = \{\{q_1, q_3\}\}$

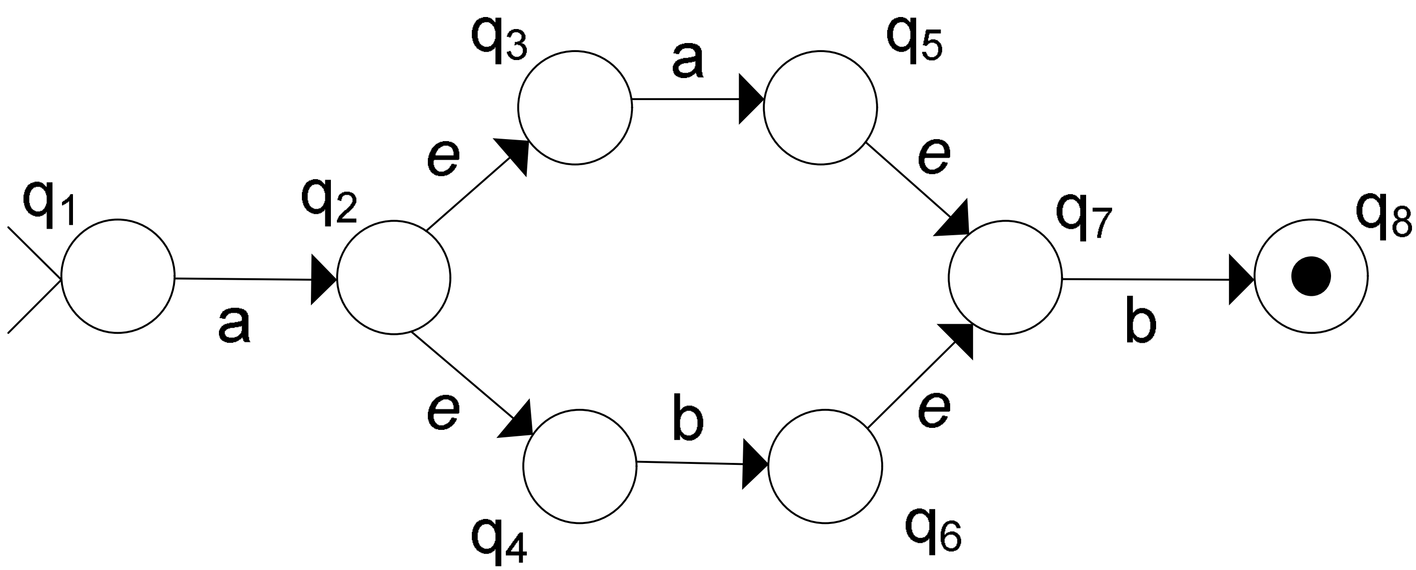


NFA → DFA: Example 2



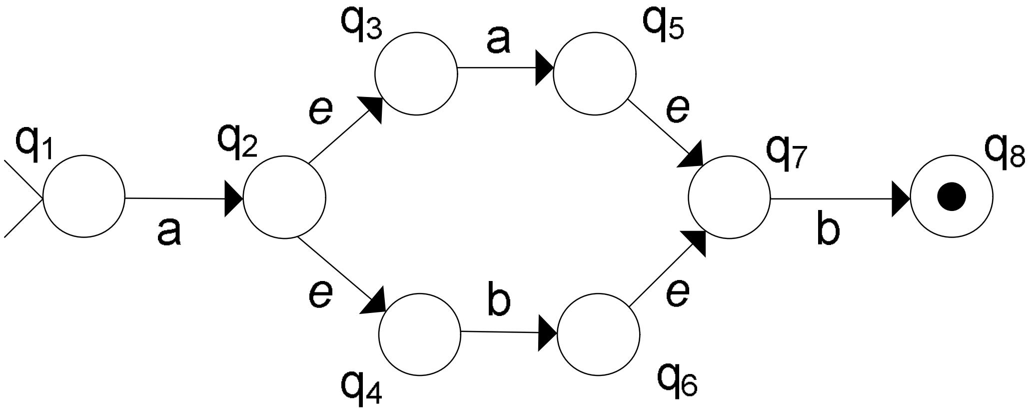
NFA → DFA: Example 3

- Give DFA M' such that L(M') = L(M) !
  - M is given below



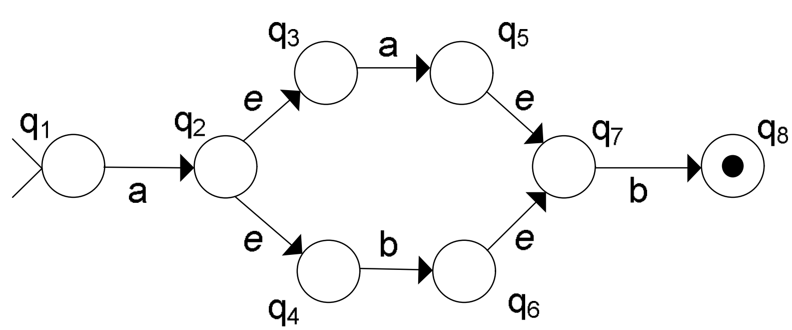
NFA → DFA: Example 3

- Definition of E-sets:
  - $E(q_1) = \{q_1\}$
  - $E(q_2) = \{q_2, q_3, q_4\}$
  - $E(q_3) = \{q_3\}$
  - $E(q_4) = \{q_4\}$
  - $E(q_5) = \{q_5, q_7\}$
  - $E(q_6) = \{q_6, q_7\}$
  - $E(q_7) = \{q_7\}$
  - $E(q_8) = \{q_8\}$



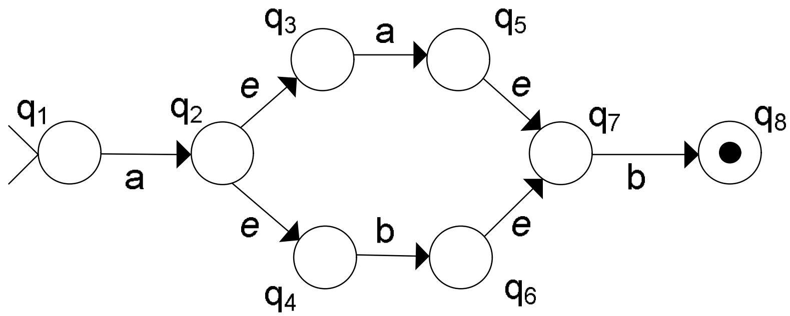
NFA → DFA: Example 3

- Definition of new initial state
  - $s_3 = E(q_1) = \{q_1\} = Q_1$
- Definition of transition function:
  - $\delta(\{q_1\}, a) = E(q_2) = \{q_2, q_3, q_4\} = Q_2$
  - $\delta(\{q_1\}, b) = \varnothing = Q_3$
  - $\delta(\{q_2, q_3, q_4\}, a) = E(q_5) = \{q_5, q_7\} = Q_4$
  - $\delta(\{q_2, q_3, q_4\}, b) = E(q_6) = \{q_6, q_7\} = Q_5$
  - $\delta(\varnothing, a) = \varnothing = Q_3$
  - $\delta(\varnothing, b) = \varnothing = Q_3$



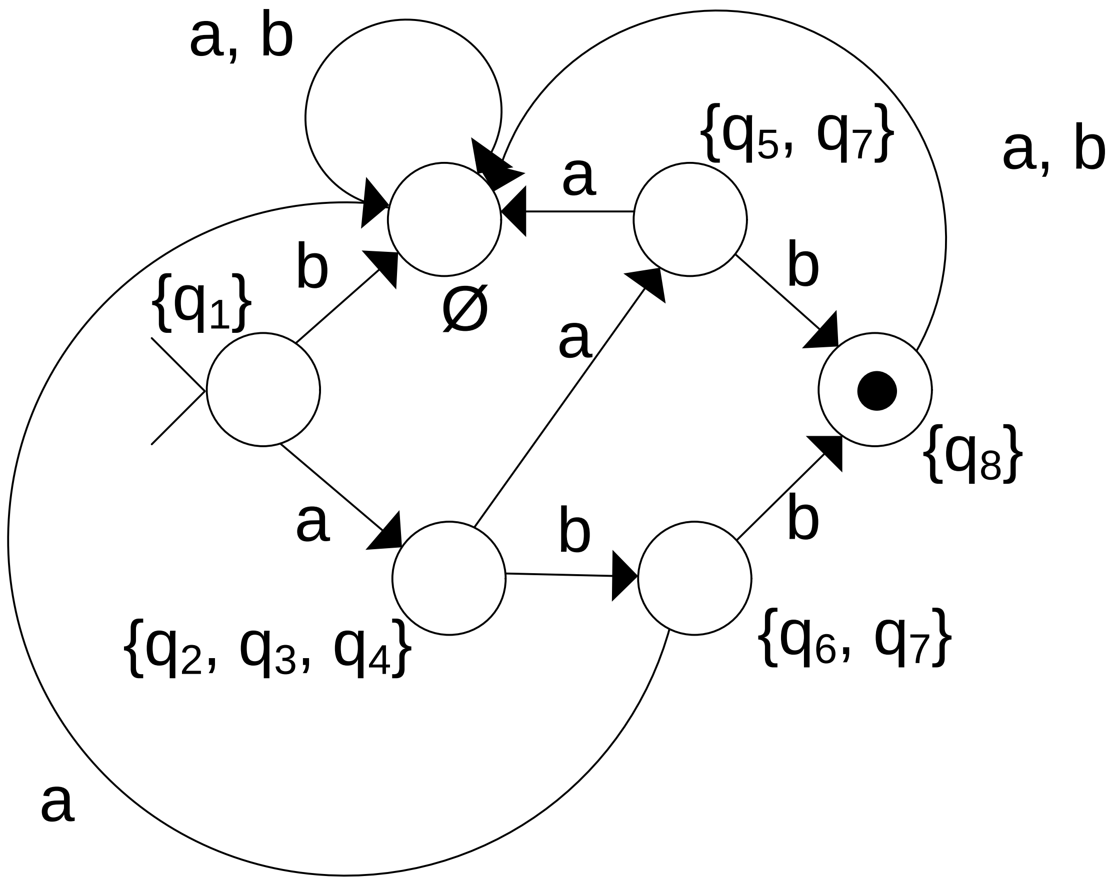
NFA → DFA: Example 3

- $\delta(\{q_5, q_7\}, a) = \varnothing = Q_3$
- $\delta(\{q_5, q_7\}, b) = E(q_8) = \{q_8\} = Q_6$
- $\delta(\{q_6, q_7\}, a) = \varnothing = Q_3$
- $\delta(\{q_6, q_7\}, b) = E(q_8) = \{q_8\} = Q_6$
- $\delta(\{q_8\}, a) = \varnothing = Q_3$
- $\delta(\{q_8\}, b) = \varnothing = Q_3$



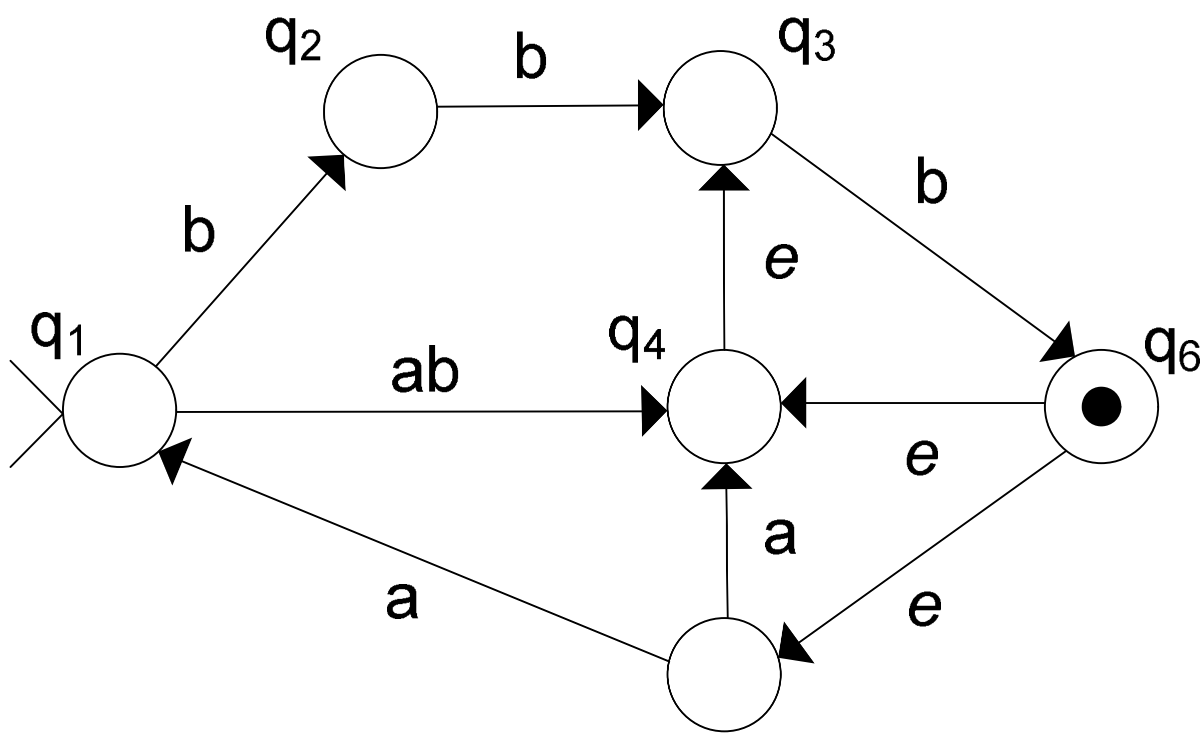
• Definition of final states
- $F_3 = \{\{q_8\}\}$

NFA → DFA: Example 3



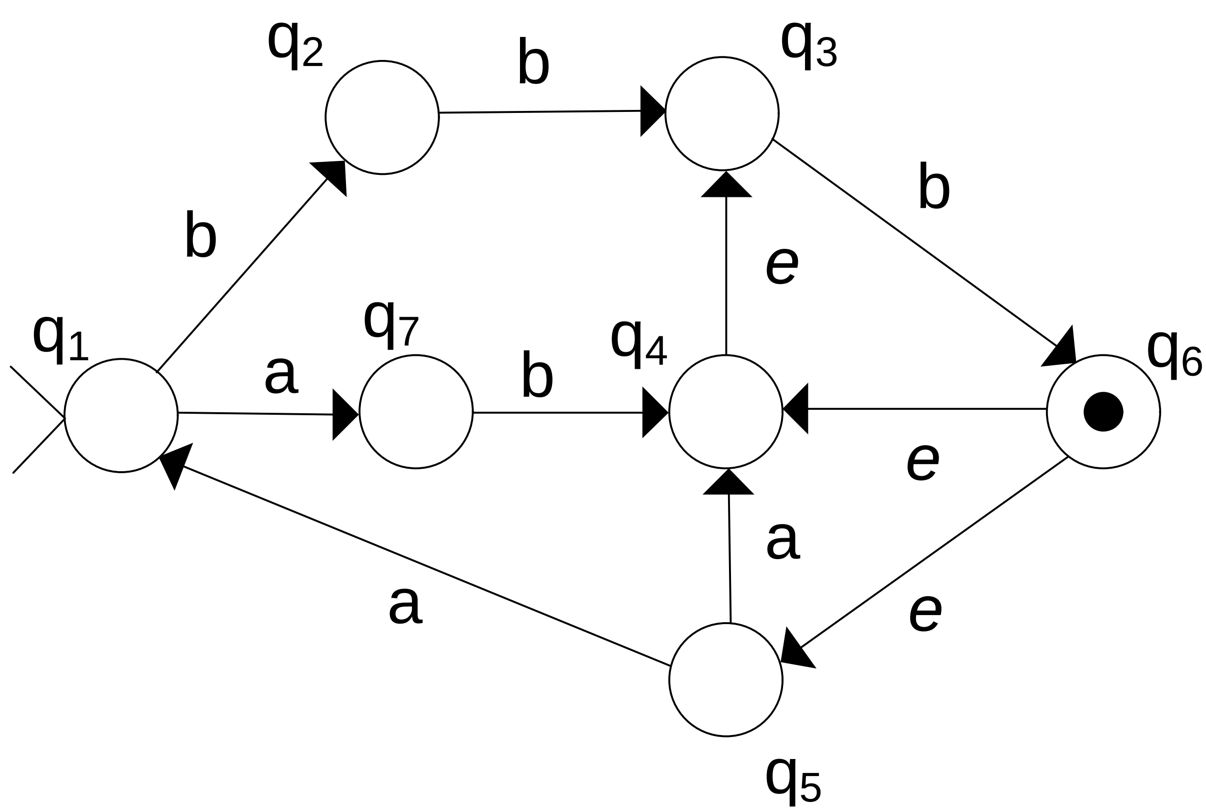
NFA → DFA: Example 4

- Give DFA M' such that L(M') = L(M) !
  - M is given below



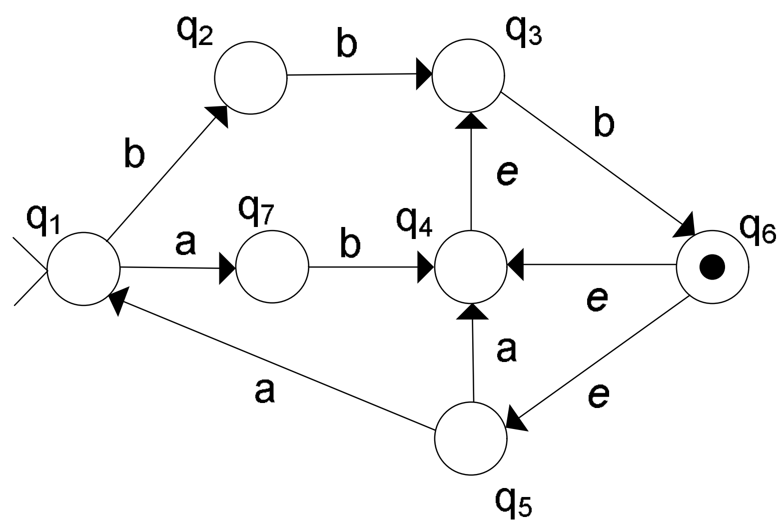
NFA → DFA: Example 4

- Eliminating string transitions by introducing new states



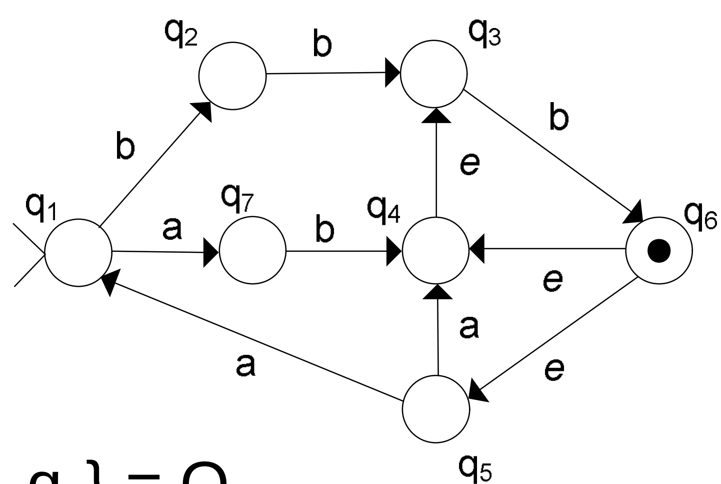
NFA → DFA: Example 4

- Definition of E-sets:
  - $E(q_1) = \{q_1\}$
  - $E(q_2) = \{q_2\}$
  - $E(q_3) = \{q_3\}$
  - $E(q_4) = \{q_3, q_4\}$
  - $E(q_5) = \{q_5\}$
  - $E(q_6) = \{q_3, q_4, q_5, q_6\}$
  - $E(q_7) = \{q_7\}$

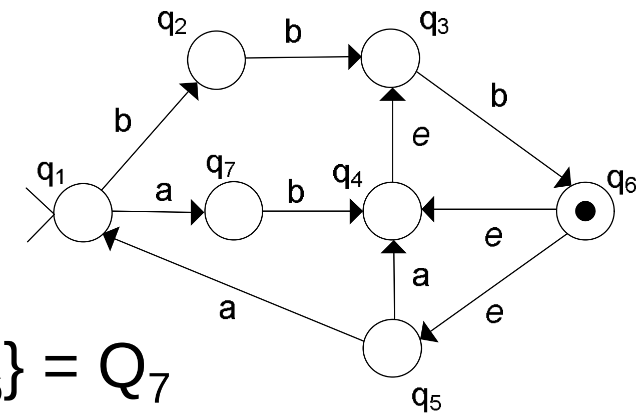


NFA → DFA: Example 4

- Definition of new initial state
  - $s_3 = E(q_1) = \{q_1\} = Q_1$
- Definition of transition function:
  - $\delta(\{q_1\}, a) = E(q_7) = \{q_7\} = Q_2$
  - $\delta(\{q_1\}, b) = E(q_2) = \{q_2\} = Q_3$
  - $\delta(\{q_7\}, a) = \varnothing = Q_4$
  - $\delta(\{q_7\}, b) = E(q_3) \cup E(q_4) = \{q_3, q_4\} = Q_5$
  - $\delta(\{q_2\}, a) = \varnothing = Q_4$
  - $\delta(\{q_2\}, b) = E(q_3) = \{q_3\} = Q_6$
  - $\delta(\varnothing, a) = \varnothing = Q_4$
  - $\delta(\varnothing, b) = \varnothing = Q_4$



NFA → DFA: Example 4



– $\delta(\{q_3, q_4\}, a) = \emptyset = Q_4$

– $\delta(\{q_3, q_4\}, b) = E(q_6) = \{q_3, q_4, q_5, q_6\} = Q_7$

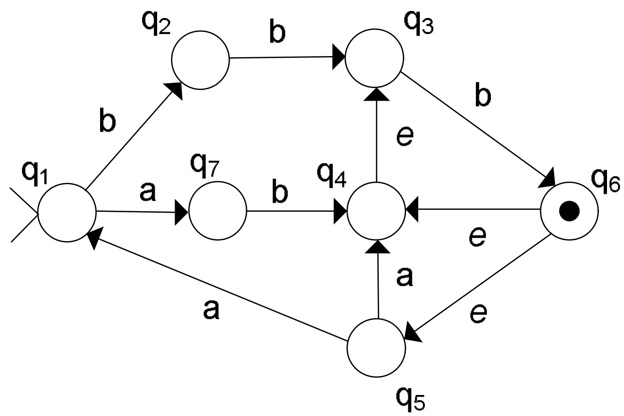
– $\delta(\{q_3\}, a) = \emptyset = Q_4$

– $\delta(\{q_3\}, b) = E(q_6) = \{q_3, q_4, q_5, q_6\} = Q_7$

– $\delta(\{q_3, q_4, q_5, q_6\}, a) = E(q_1) \cup E(q_4) = \{q_1, q_3, q_4\} = Q_8$

– $\delta(\{q_3, q_4, q_5, q_6\}, b) = E(q_6) = \{q_3, q_4, q_5, q_6\} = Q_7$

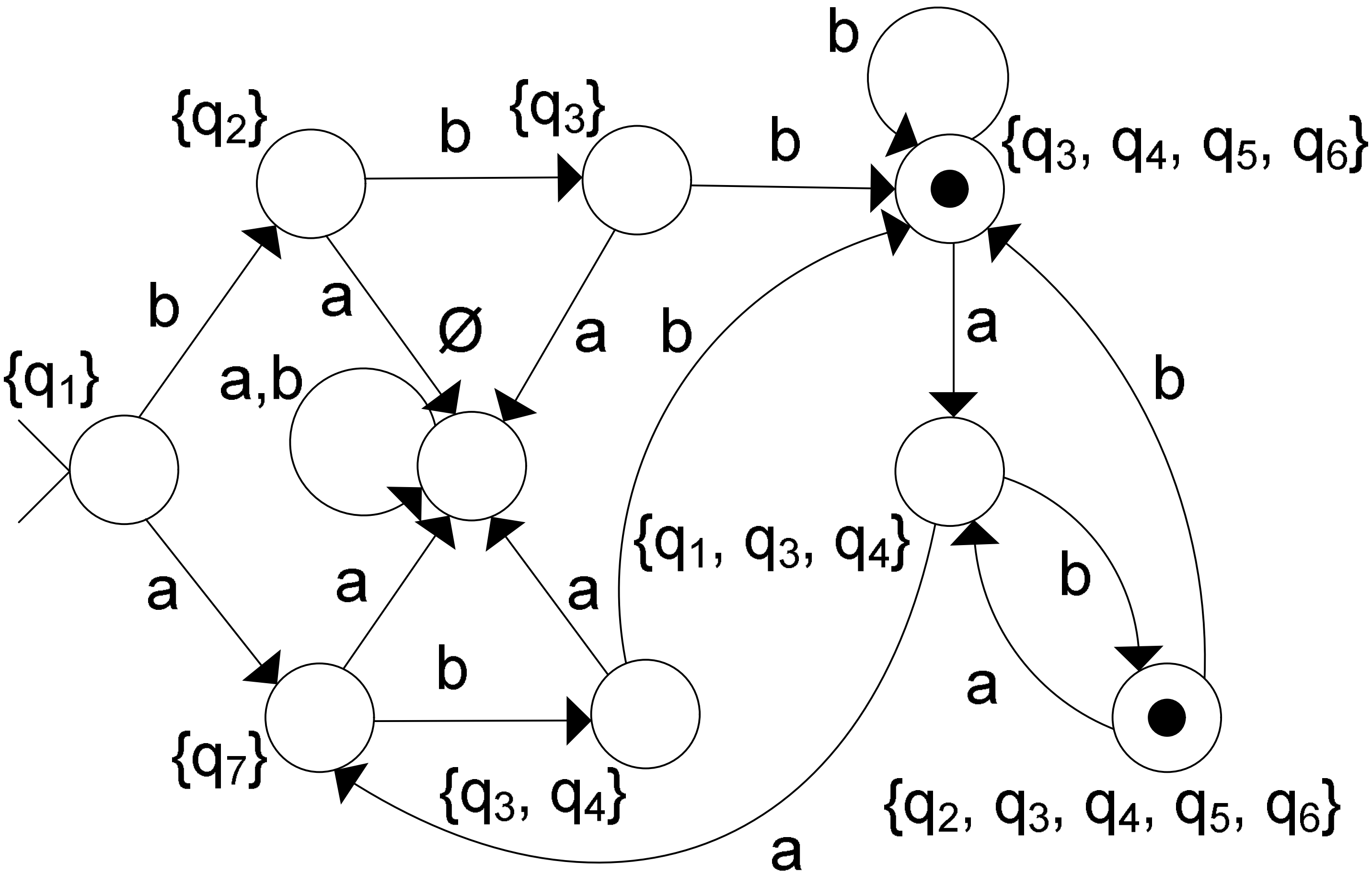
NFA → DFA: Example 4



- $\delta(\{q_1, q_3, q_4\}, a) = E(q_7) = \{q_7\} = Q_2$
- $\delta(\{q_1, q_3, q_4\}, b) =$
  $E(q_2) \cup E(q_6) =$
  $\{q_2, q_3, q_4, q_5, q_6\} = Q_9$
- $\delta(\{q_2, q_3, q_4, q_5, q_6\}, a) = E(q_1) \cup E(q_4) =$
  $\{q_1, q_3, q_4\} = Q_8$
- $\delta(\{q_2, q_3, q_4, q_5, q_6\}, b) =$
  $E(q_3) \cup E(q_6) = \{q_3, q_4, q_5, q_6\} = Q_7$

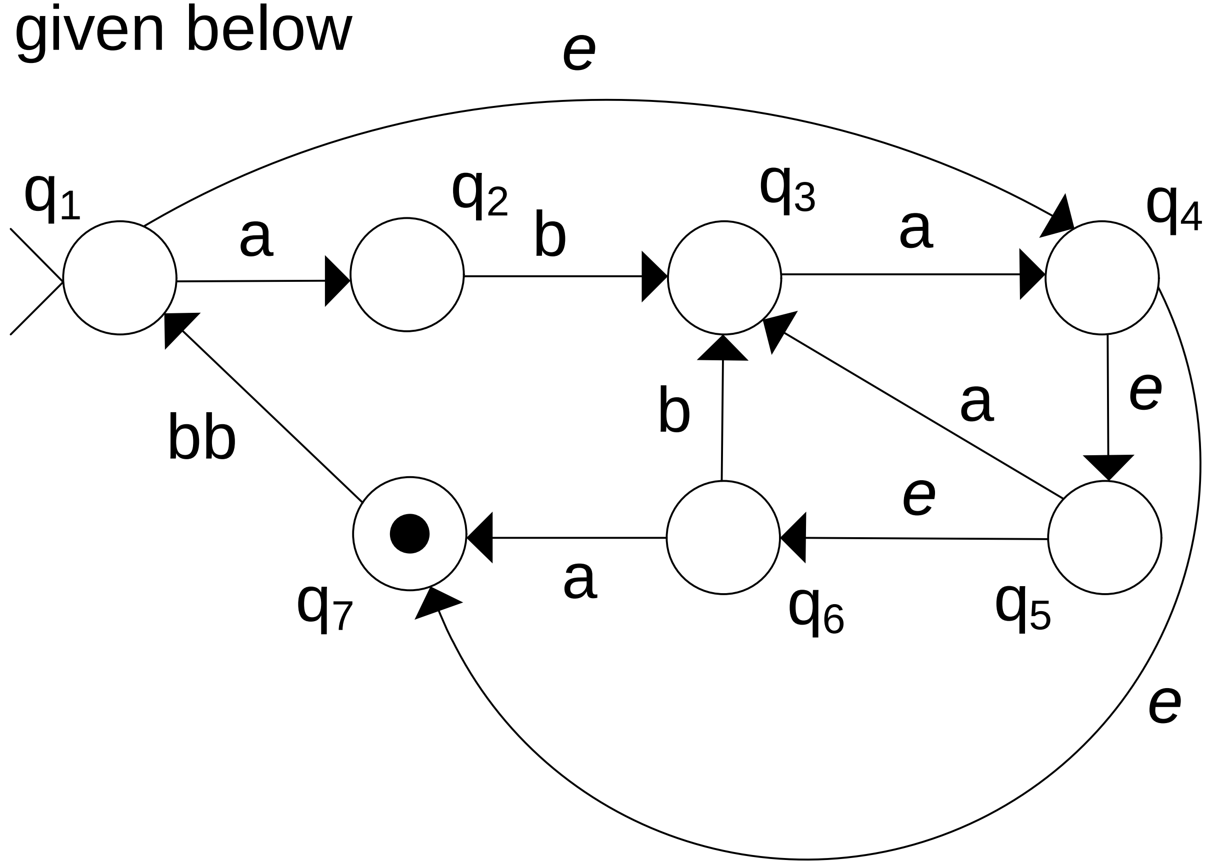
- Definition of final states
  - $F_3 = \{\{q_3, q_4, q_5, q_6\}, \{q_2, q_3, q_4, q_5, q_6\}\}$

NFA → DFA: Example 4



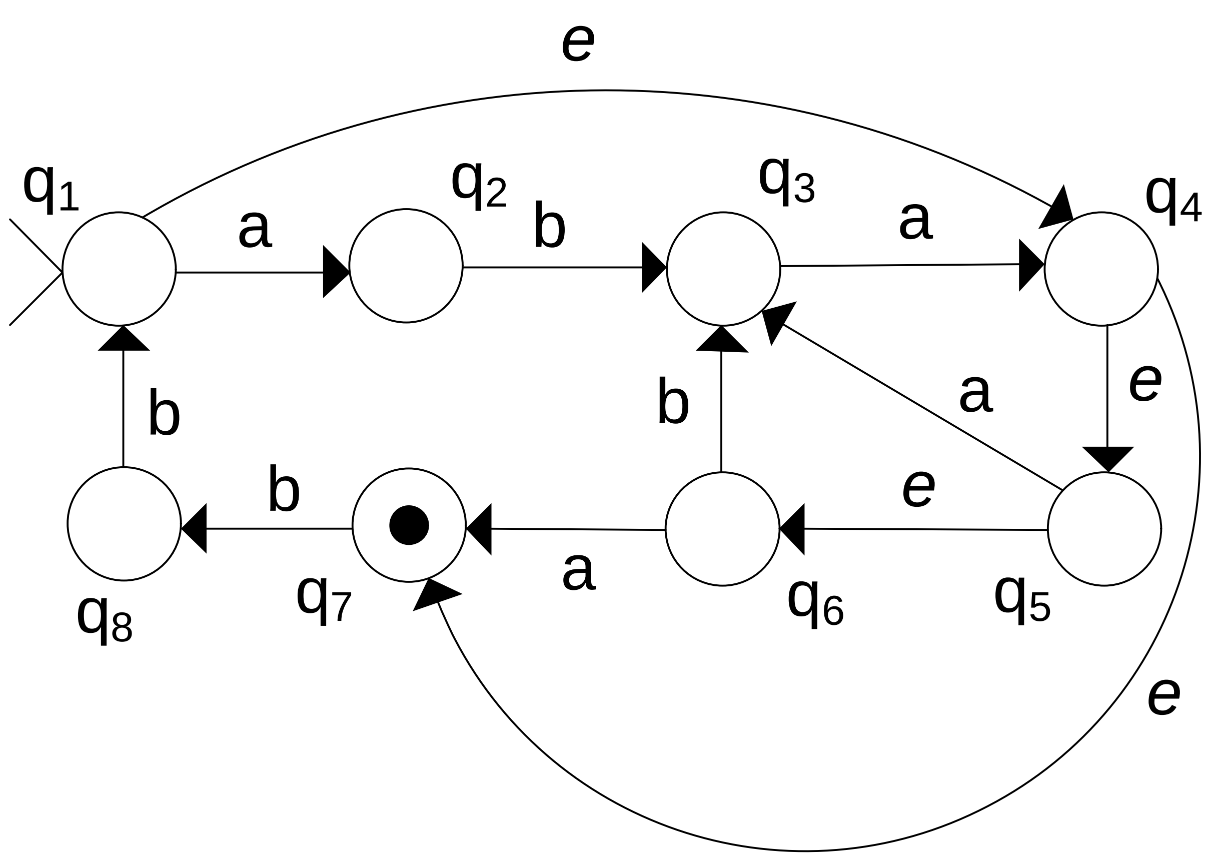
NFA → DFA: Example 5

- Give DFA M' such that L(M') = L(M) !
  - M is given below



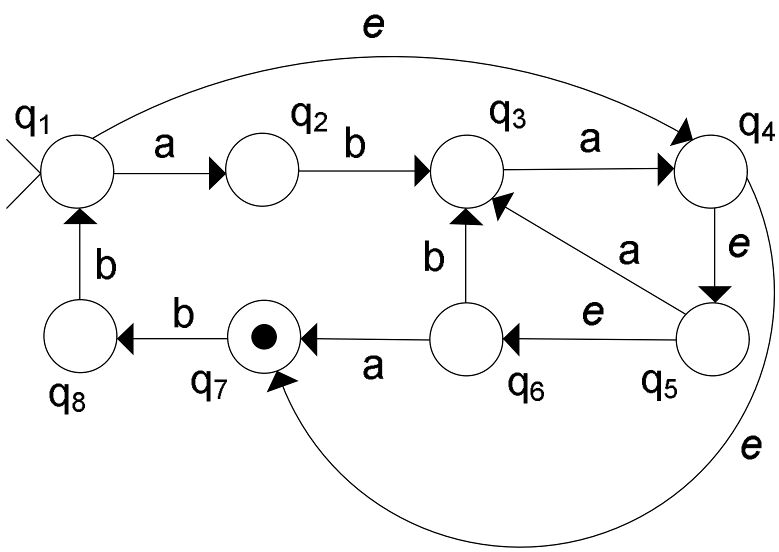
NFA → DFA: Example 5

- Eliminating string transitions by introducing new states

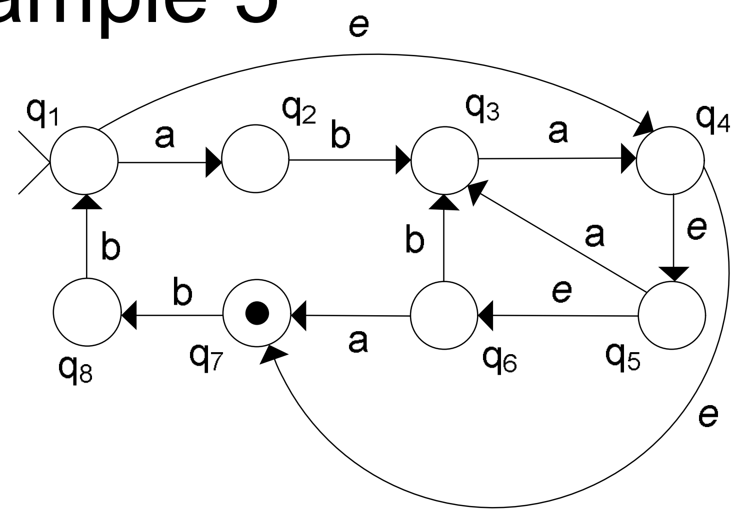


NFA → DFA: Example 5

- Definition of E-sets:
  - $E(q_1) = \{q_1, q_4, q_5, q_6, q_7\}$
  - $E(q_2) = \{q_2\}$
  - $E(q_3) = \{q_3\}$
  - $E(q_4) = \{q_4, q_5, q_6, q_7\}$
  - $E(q_5) = \{q_5, q_6\}$
  - $E(q_6) = \{q_6\}$
  - $E(q_7) = \{q_7\}$
  - $E(q_8) = \{q_8\}$

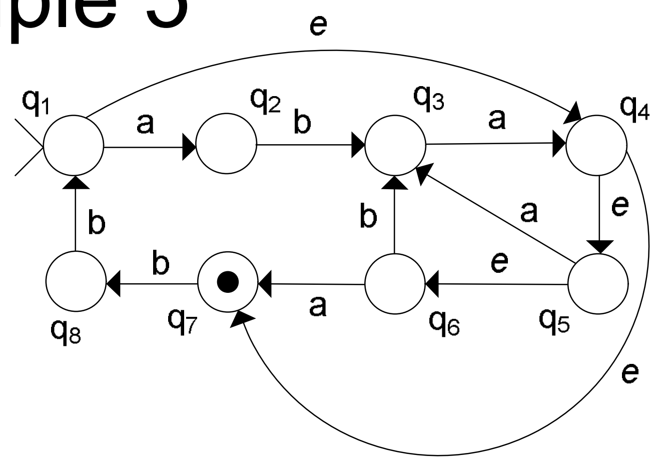


NFA → DFA: Example 5



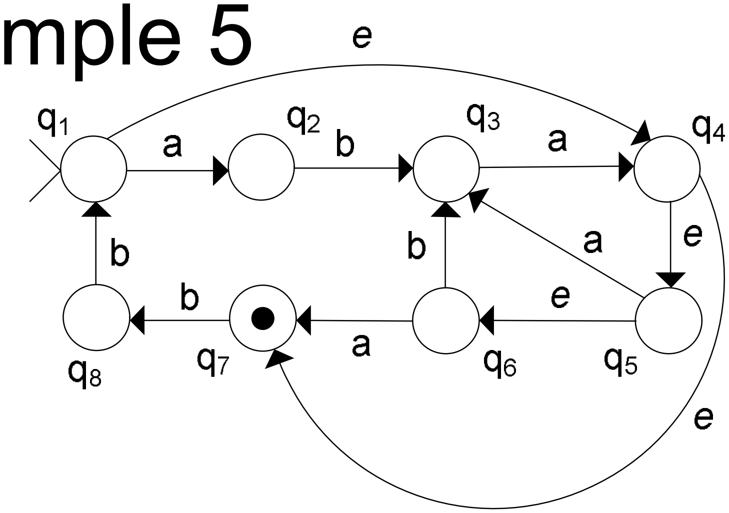
- Definition of new initial state
  - $s_3 = E(q_1) = \{q_1, q_4, q_5, q_6, q_7\} = Q_1$
- Definition of transition function:
  - $\delta(\{q_1, q_4, q_5, q_6, q_7\}, a) = E(q_2) \cup E(q_3) \cup E(q_7) = \{q_2, q_3, q_7\} = Q_2$
  - $\delta(\{q_1, q_4, q_5, q_6, q_7\}, b) = E(q_3) \cup E(q_8) = \{q_3, q_8\} = Q_3$

NFA → DFA: Example 5



- $\delta(\{q_2, q_3, q_7\}, a) = E(q_4) =$
  $\{q_4, q_5, q_6, q_7\} = Q_4$
- $\delta(\{q_2, q_3, q_7\}, b) =$
  $E(q_3) \cup E(q_8) = \{q_3, q_8\} = Q_3$
- $\delta(\{q_3, q_8\}, a) = E(q_4) = \{q_4, q_5, q_6, q_7\} = Q_4$
- $\delta(\{q_3, q_8\}, b) = E(q_1) = \{q_1, q_4, q_5, q_6, q_7\} = Q_1$
- $\delta(\{q_4, q_5, q_6, q_7\}, a) =$
  $E(q_3) \cup E(q_7) = \{q_3, q_7\} = Q_5$
- $\delta(\{q_4, q_5, q_6, q_7\}, b) = E(q_3) \cup E(q_8) = \{q_3, q_8\} = Q_3$

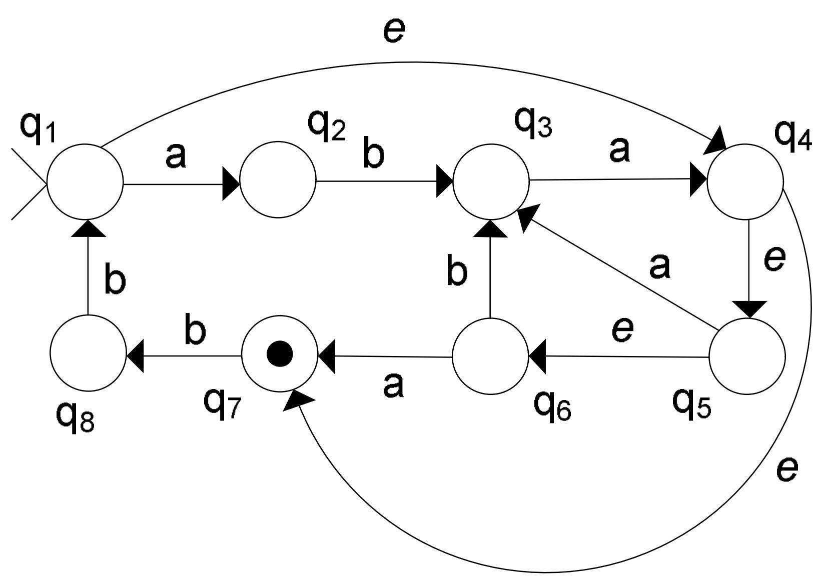
NFA → DFA: Example 5



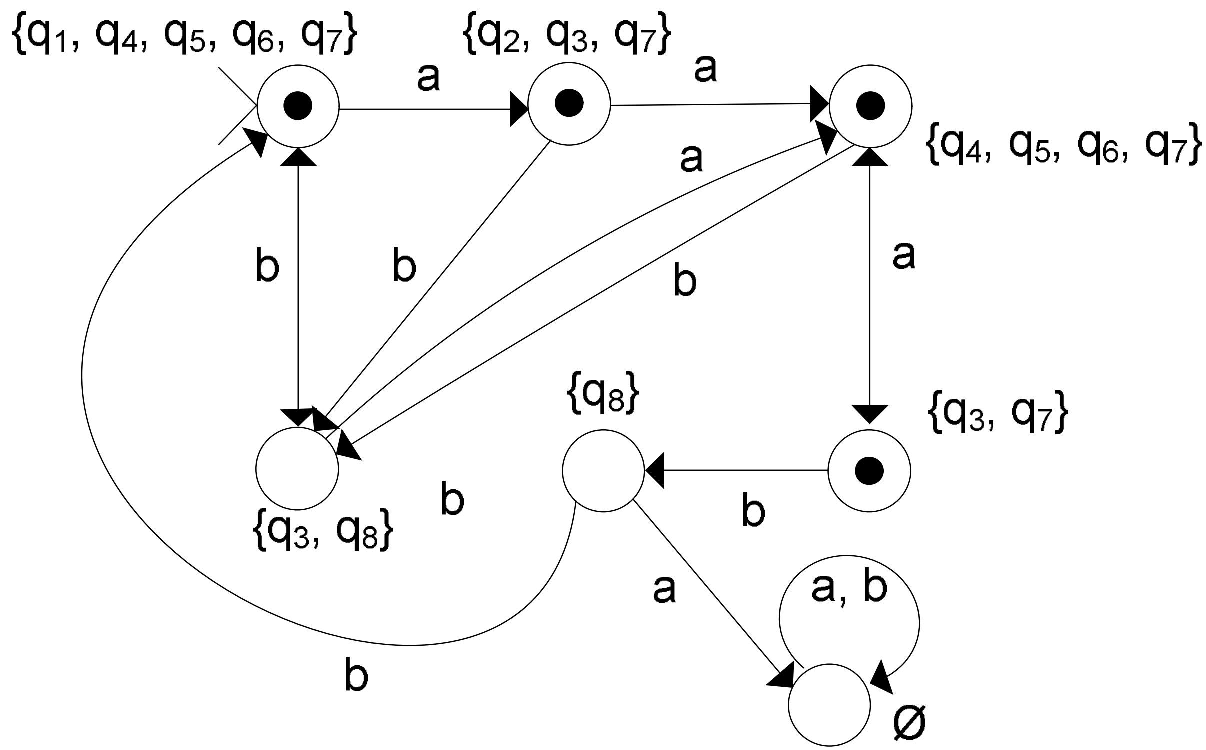
- $\delta(\{q_3, q_7\}, a) =$
  $E(q_4) = \{q_4, q_5, q_6, q_7\} = Q_4$
- $\delta(\{q_3, q_7\}, b) = E(q_8) = \{q_8\} = Q_6$
- $\delta(\{q_8\}, a) = \emptyset = Q_7$
- $\delta(\{q_8\}, b) = E(q_1) = \{q_1, q_4, q_5, q_6, q_7\} = Q_1$
- $\delta(\{\}, a) = \emptyset = Q_7$
- $\delta(\{\}, b) = \emptyset = Q_7$

NFA → DFA: Example 5

- Definition of final states
  - $F_3 = \{\{q_1, q_4, q_5, q_6, q_7\}, \{q_2, q_3, q_7\}, \{q_4, q_5, q_6, q_7\}, \{q_3, q_7\}\}$



NFA → DFA: Example 5

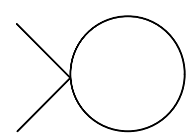


Element of the Theory of the Computation

Lecture 6

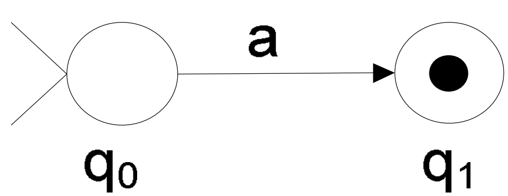
RE → NFA: Example 1

- Define NFA M, where L(M) = L(R) and R = Ø !



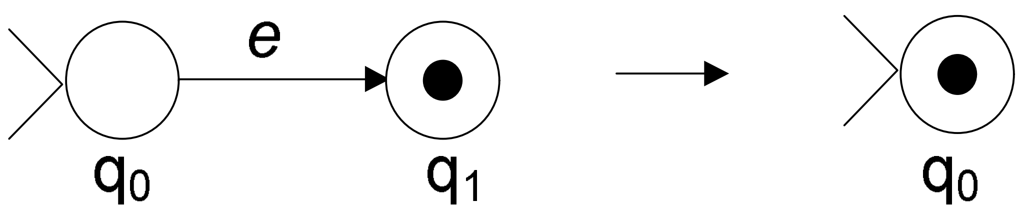
RE → NFA: Example 2

- Define NFA M, where L(M) = L(R) and R = a !



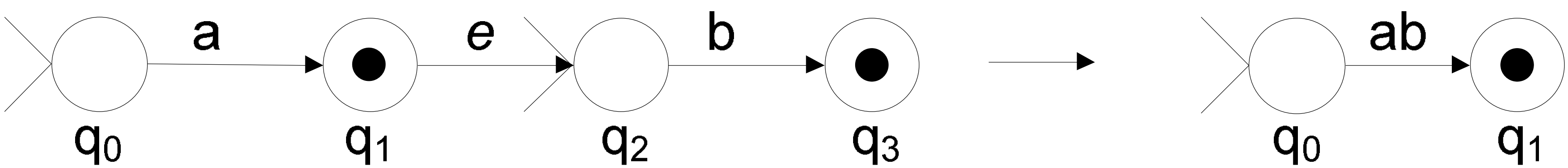
RE → NFA: Example 3

- Define NFA M, where L(M) = L(R) and R = Ø* !



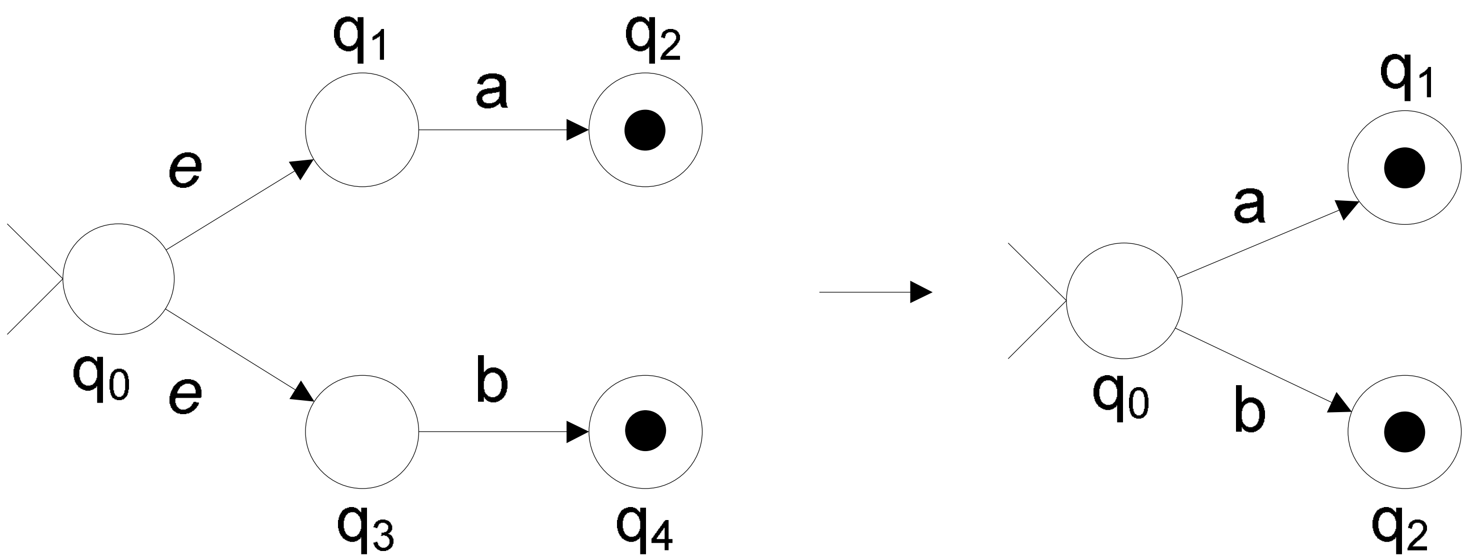
RE → NFA: Example 4

- Define NFA M, where L(M) = L(R) and R = ab !



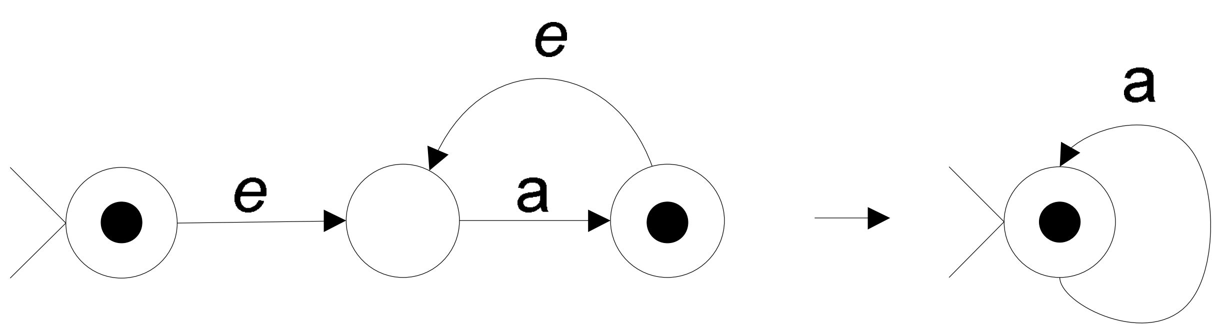
RE → NFA: Example 5

- Define NFA M, where L(M) = L(R) and R = a ∪ b !



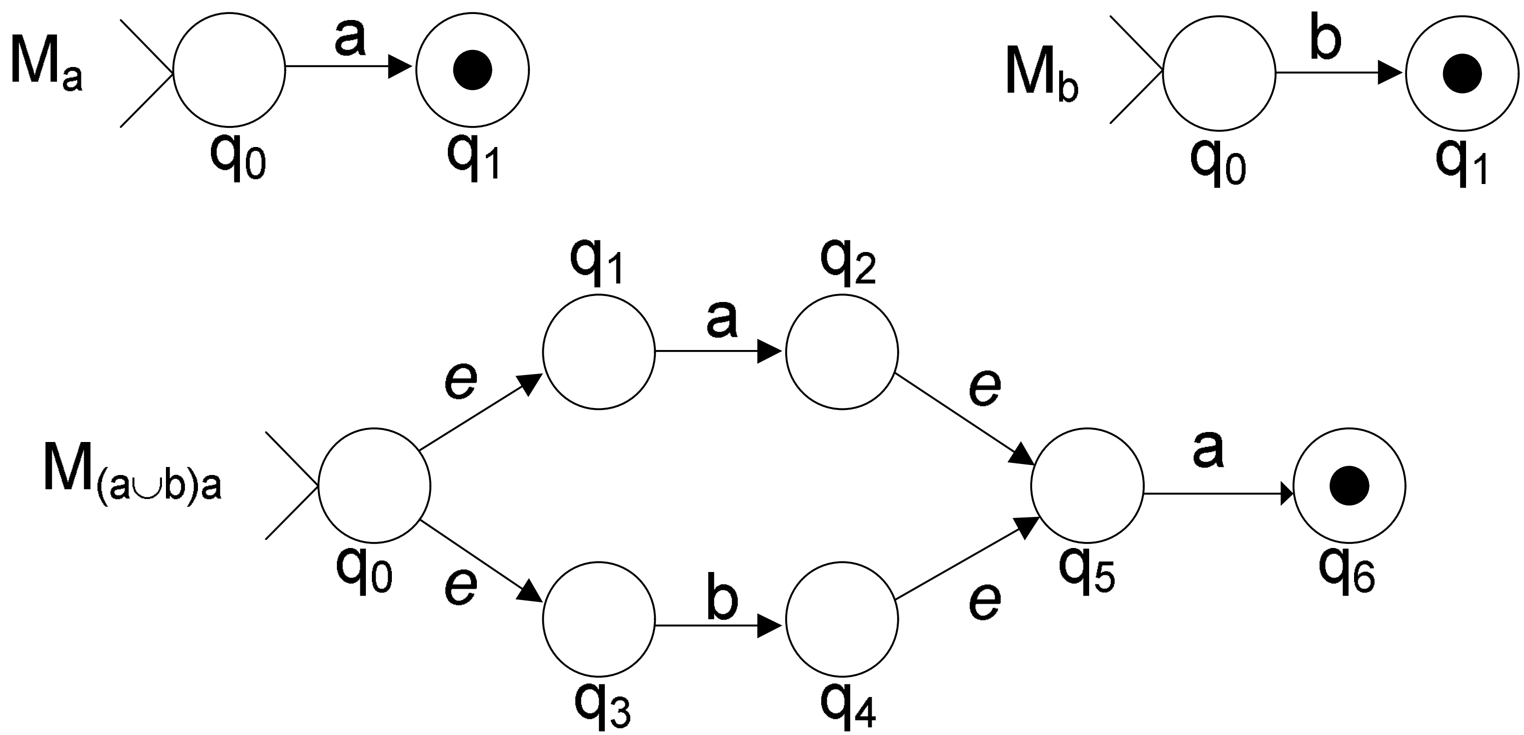
RE → NFA: Example 6

- Define NFA M, where L(M) = L(R) and R = a* !
  - simplify carefully if possible



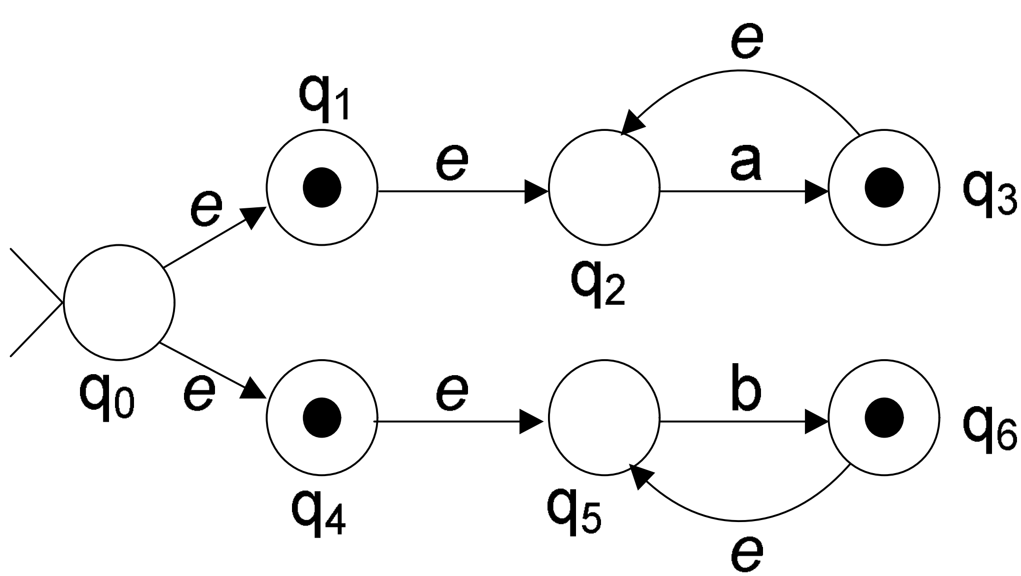
RE → NFA: Example 7

- Define NFA M, where L(M) = L(R) and R is given below! Use the construction theorems!
  - R = (a ∪ b)a



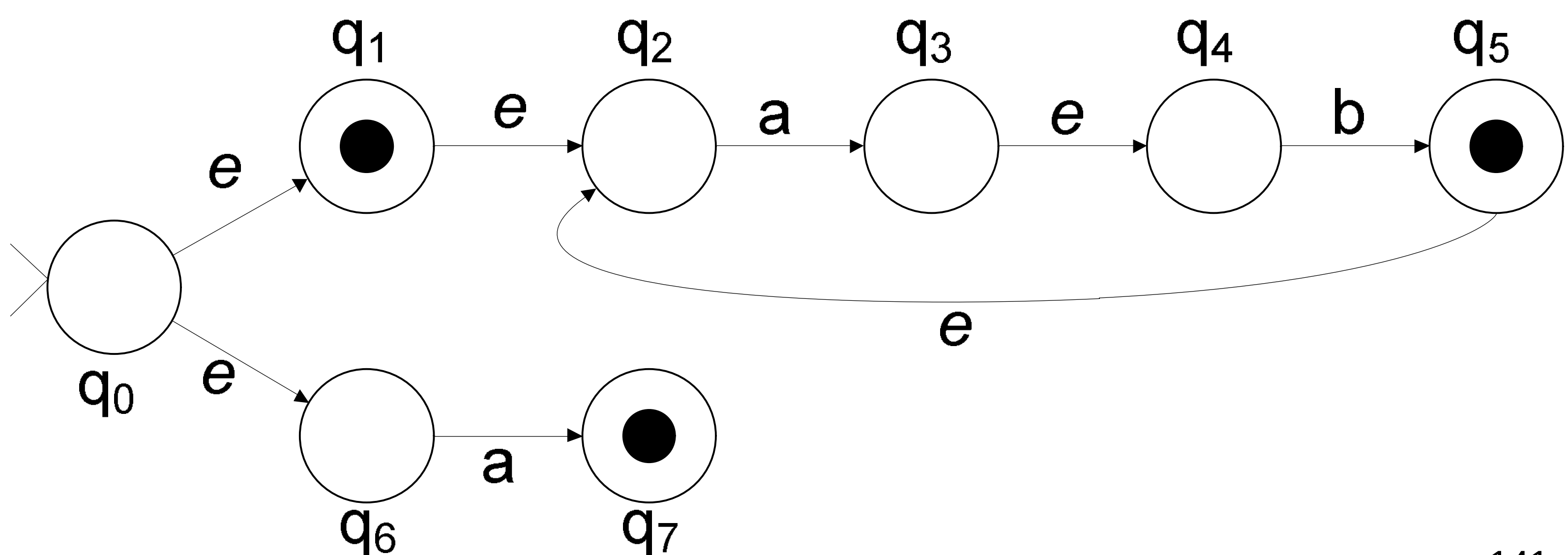
RE → NFA: Example 8

- Define NFA M, where L(M) = L(R) and R is given below! Use the construction theorems!
  - $R = a^* \cup b^*$



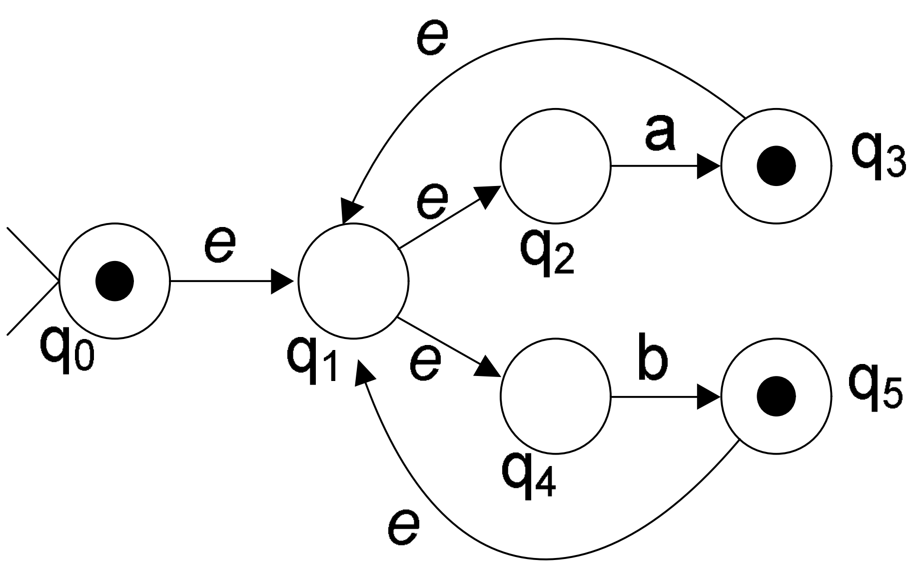
RE → NFA: Example 9

- Define NFA M, where L(M) = L(R) and R is given below! Use the construction theorems!
  – R = a ∪ (ab)*



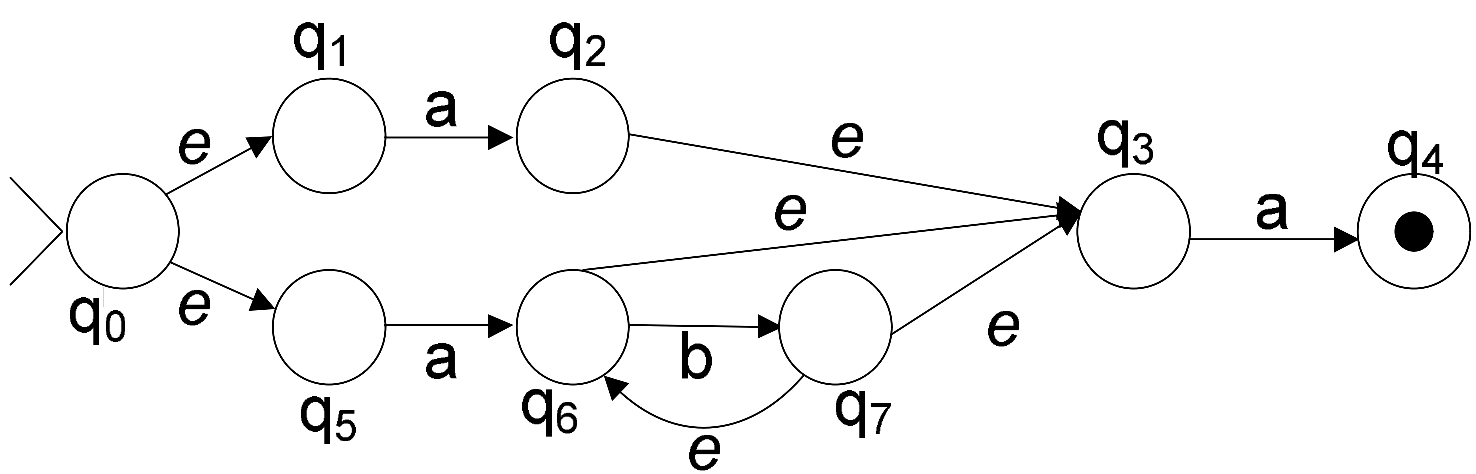
RE → NFA: Example 10

- Define NFA M, where L(M) = L(R) and R is given below! Use the construction theorems!
  - R = (a ∪ b)*



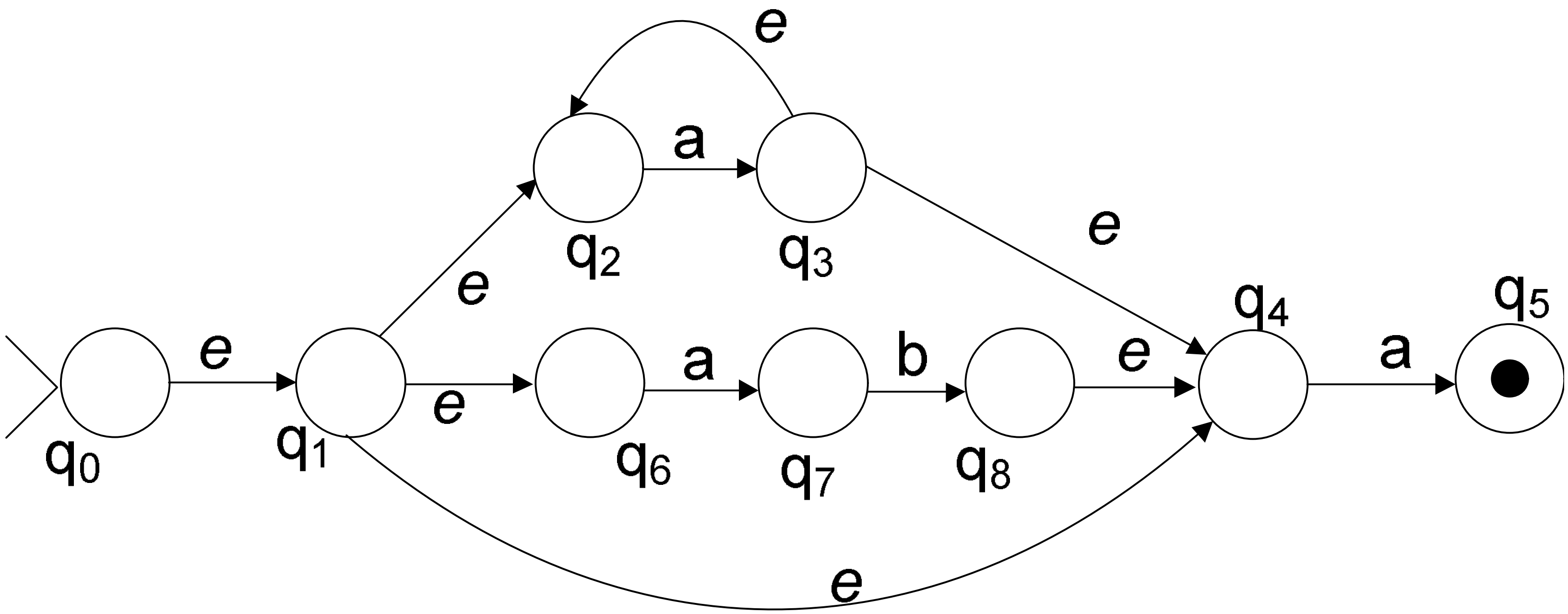
RE → NFA: Example 11

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = (a ∪ ab*)a



RE → NFA: Example 12

- Define NFA M, where L(M) = L(R) and R is given below!
  - R= $(a^* \cup ab \cup \emptyset^*)a$

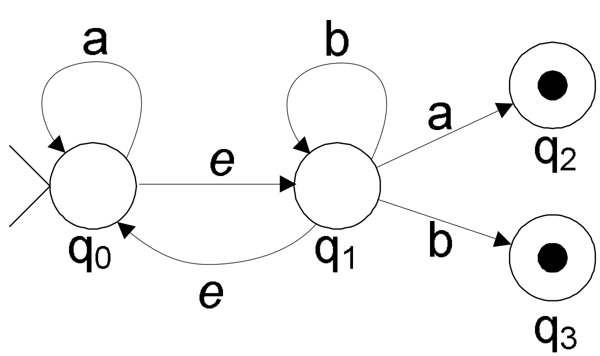


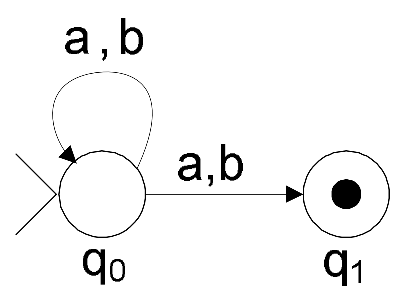
RE → NFA: Example 13

- Define NFA M, where L(M) = L(R) and R is given below!
  - $R = (a^*b^*)^*(a \cup b)$

  - $K = \{q_0, q_1, q_2, q_3\}$
  - $\Sigma = \{a, b\}$
  - $s = q_0$
  - $F = \{q_2, q_3\}$

$K = \{q_0, q_1\}$
$\Sigma = \{a, b\}$
$s = q_0$
$F = \{q_1\}$



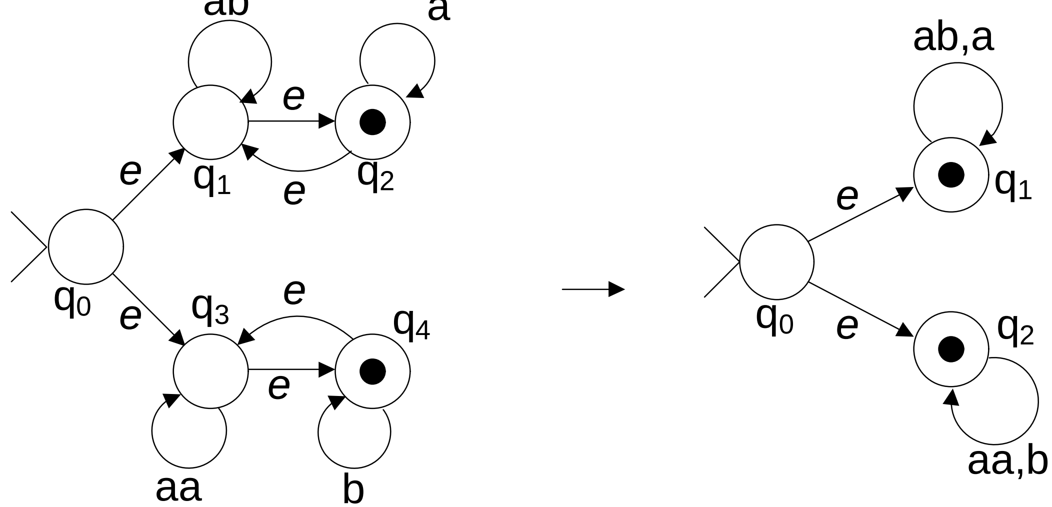


RE → NFA: Example 14

- Define NFA M, where L(M) = L(R) and R is given below!
  - $R = (((ab)^*a^*)^* \cup ((aa)^*b^*)^*)$

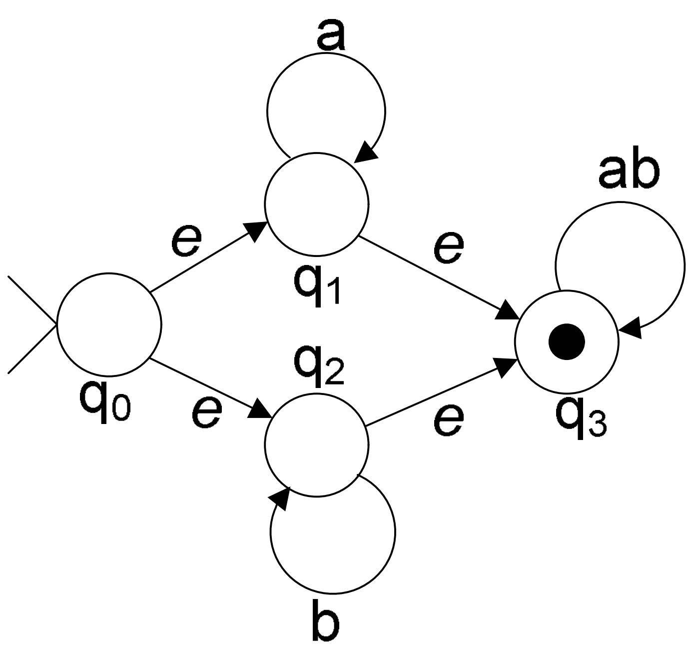
  - $K = \{q_0, q_1, q_2, q_3, q_4\}$
  - $\Sigma = \{a, b\}$
  - $s = q_0$
  - $F = \{q_2, q_4\}$

$K = \{q_0, q_1, q_2\}$
$\Sigma = \{a, b\}$
$s = q_0$
$F = \{q_1, q_2\}$



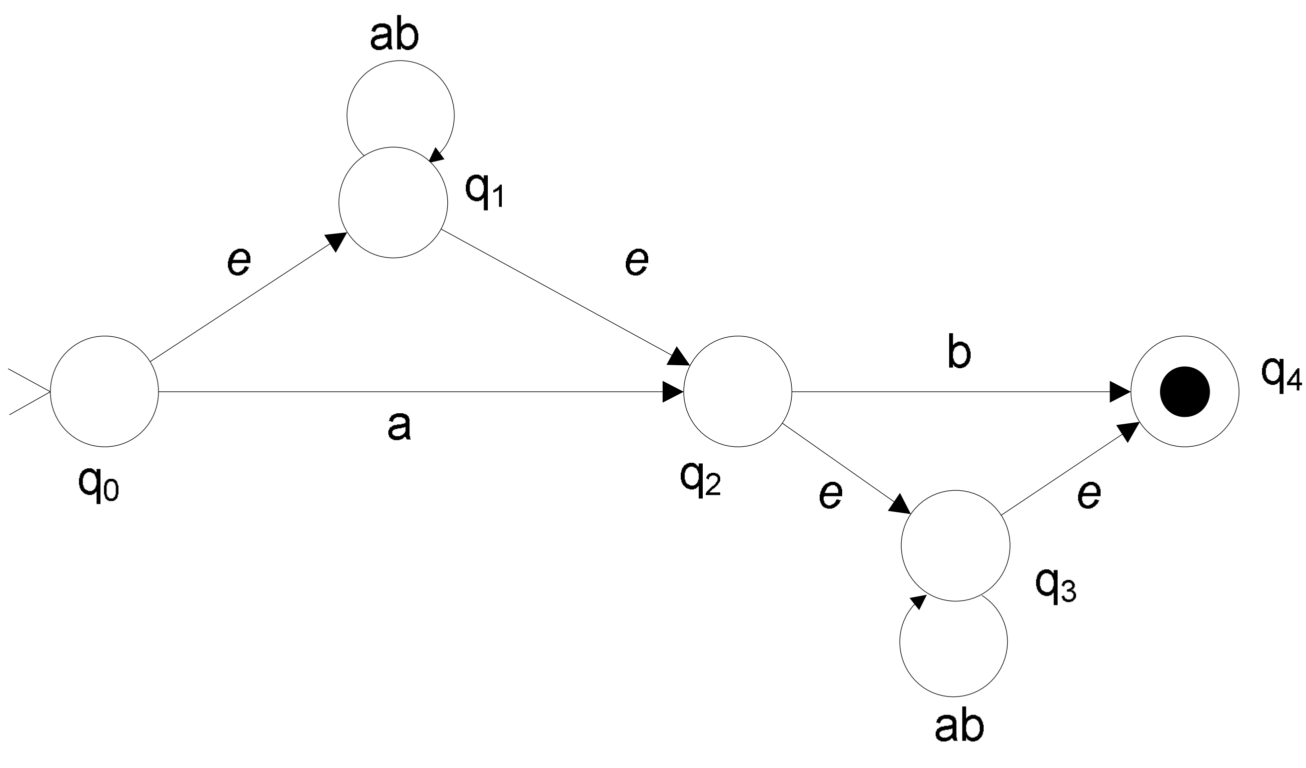
RE → NFA: Example 15

- Define NFA M, where L(M) = L(R) and R is given below!
  - $R = (a^* \cup b^*)(ab)^*$



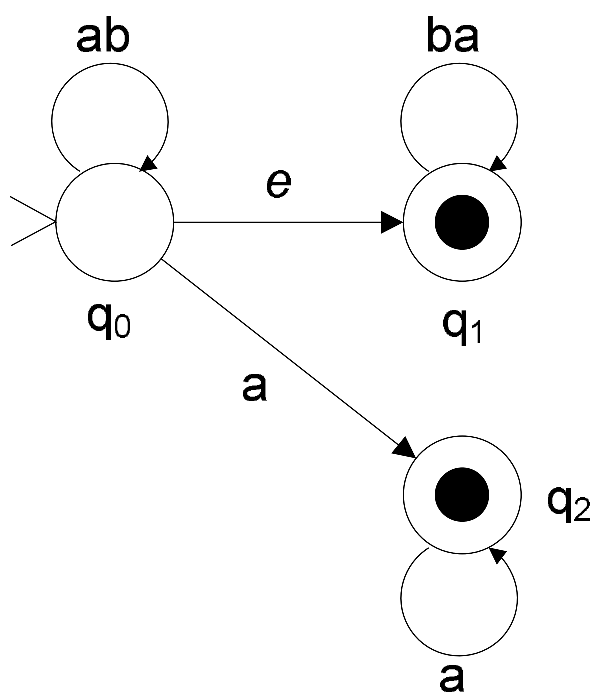
RE → NFA: Example 16

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = (a ∪ (ab)*)(b ∪ (ab)*)



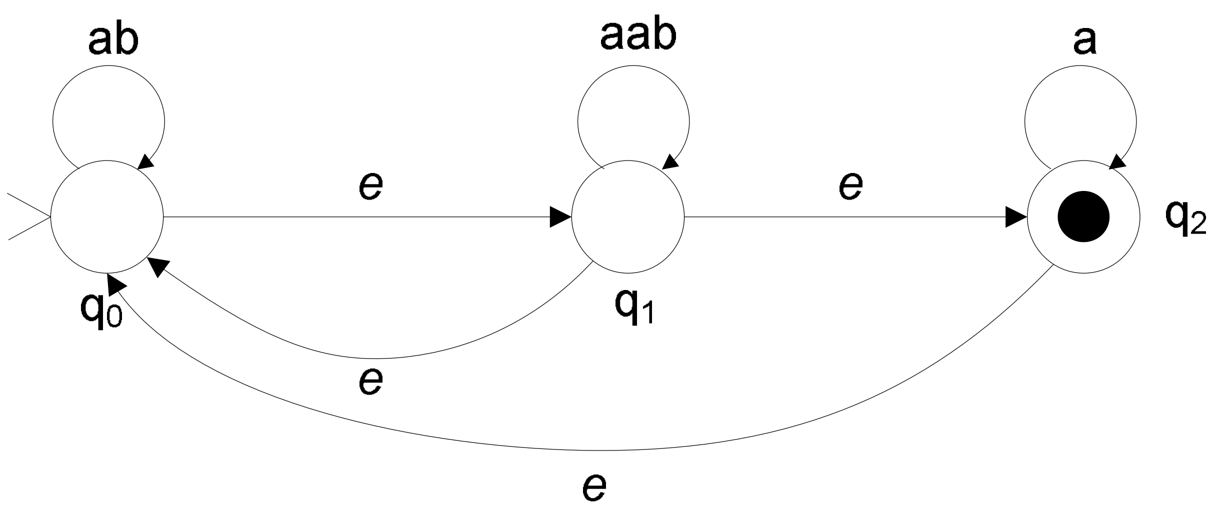
RE → NFA: Example 17

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = (ab)*((ba)* ∪ aa*)



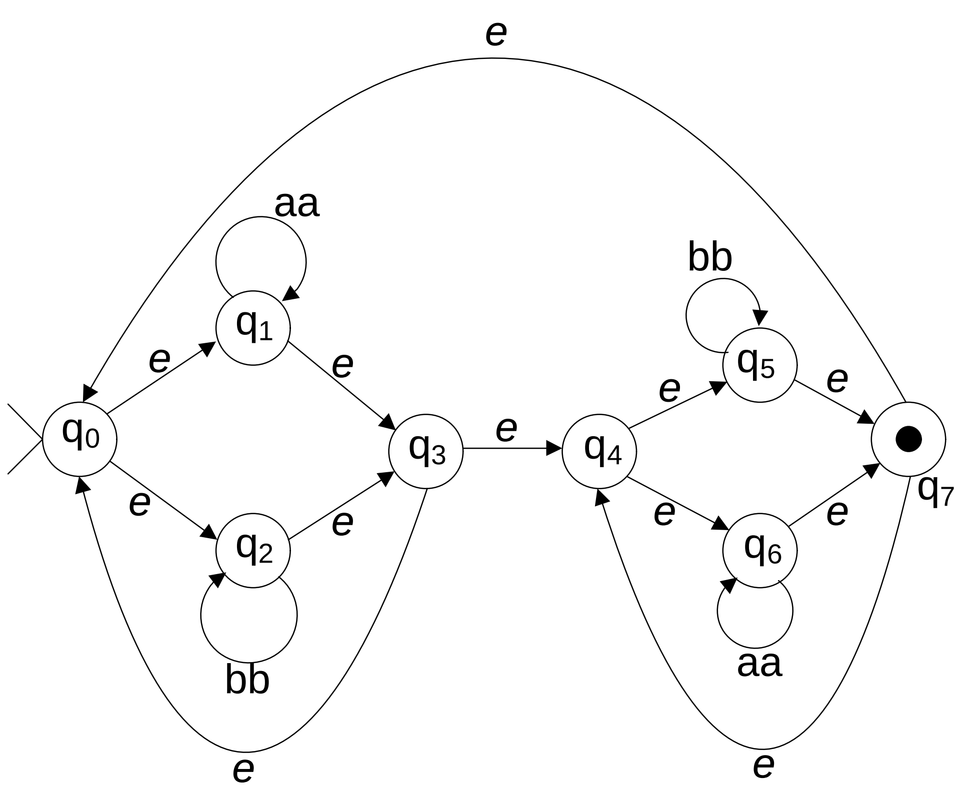
RE → NFA: Example 18

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = (((ab)*(aab)*)*a*)*



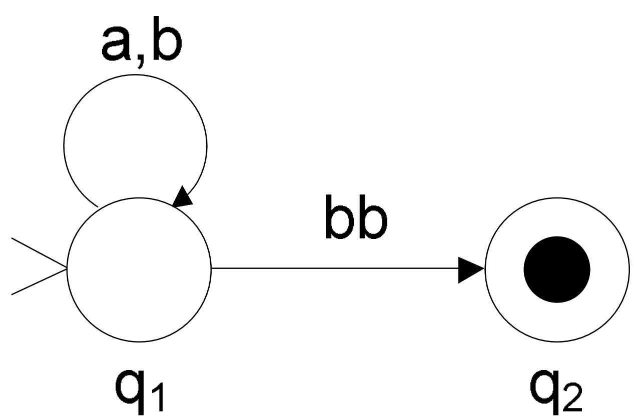
RE → NFA: Example 19

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = $(((aa)^* \cup (bb)^*)^*((bb)^* \cup (aa)^*)^*)^*$



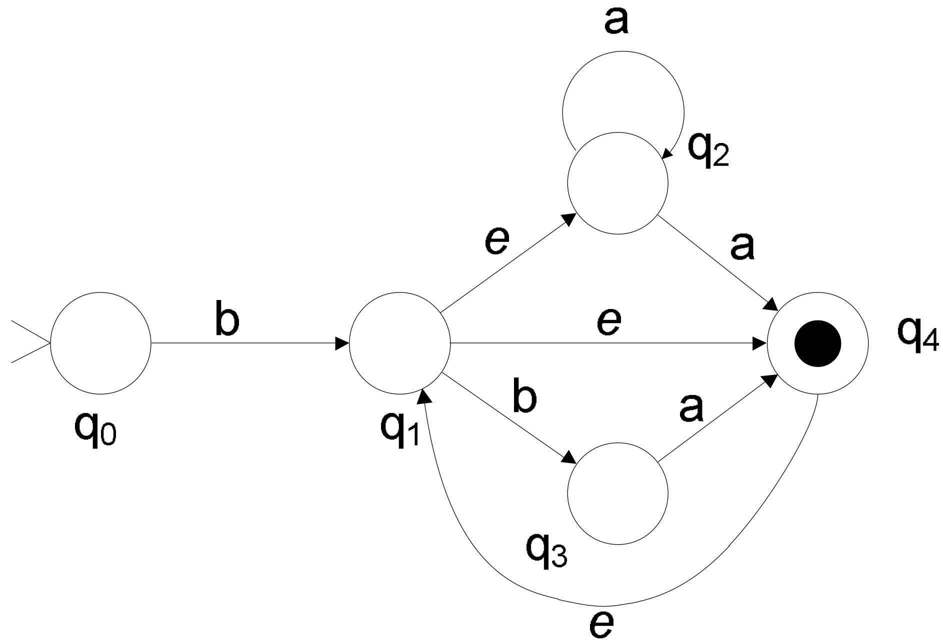
RE → NFA: Example 20

- Define NFA M, where L(M) = L(R) and R is given below!
  - $R = (a \cup b)^*bb$



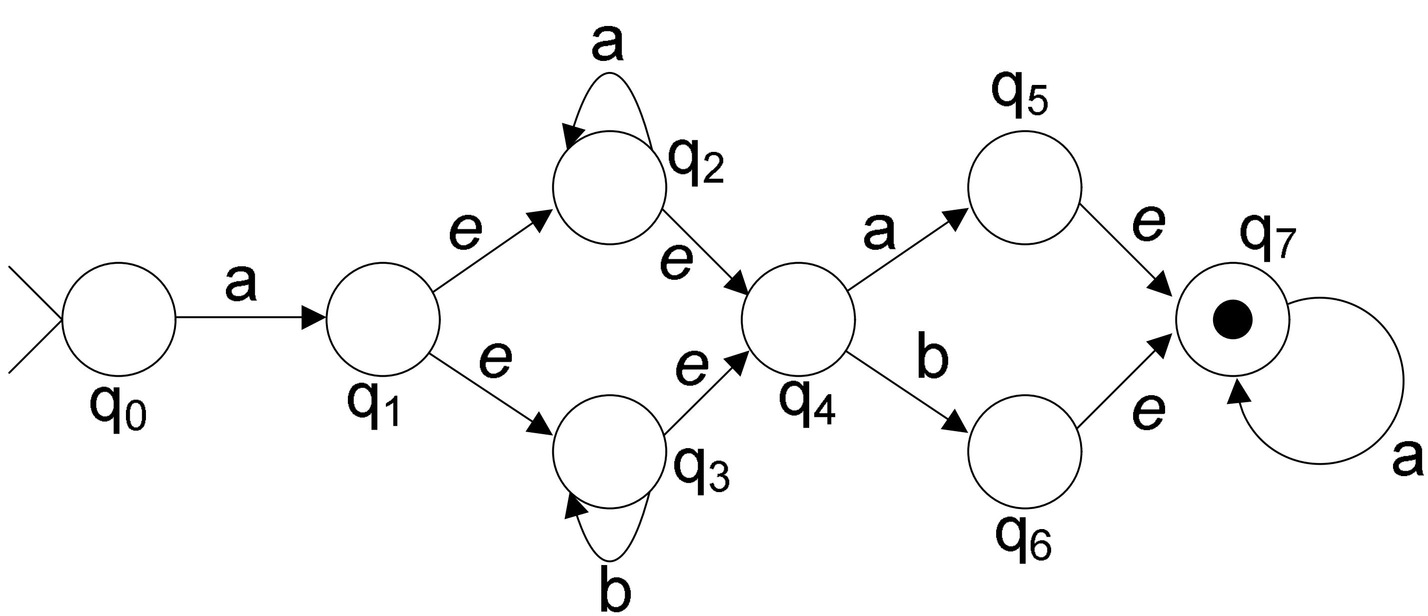
RE → NFA: Example 21

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = b((a* ∪ b)a)*



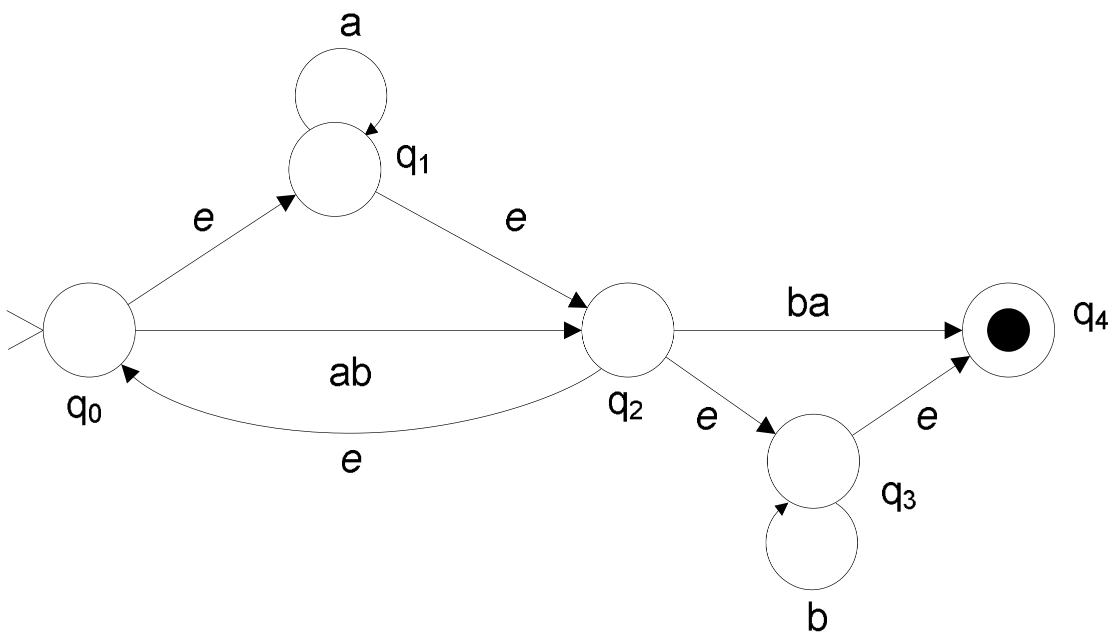
RE → NFA: Example 22

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = a(a* ∪ b*)(a ∪ b)a*



RE → NFA: Example 23

- Define NFA M, where L(M) = L(R) and R is given below!
  - R = (a* ∪ ab)*(b* ∪ ba)



Element of the Theory of the Computation

Lecture 7

CFG: Example 1

- Which words can be derived at most in 4 steps with $G = \{V, \Sigma, R, S\}$ grammar from S?
  - $V = \{a, b, A, B, S\}$
  - $\Sigma = \{a, b\}$
  - $R = \{S \rightarrow A, S \rightarrow abA, S \rightarrow aB, A \rightarrow Sa, B \rightarrow b\}$
- Solution:
  - $S \Rightarrow_G A \Rightarrow_G Sa \Rightarrow_G aBa \Rightarrow_G aba$
  - $S \Rightarrow_G abA \Rightarrow_G abSa \Rightarrow_G abaBa \Rightarrow_G ababa$
  - $S \Rightarrow_G aB \Rightarrow_G ab$

CFG: Example 2

- Which words can be derived at most in 4 steps with $G = \{V, \Sigma, R, S\}$ grammar from S?
  - $V = \{a, b, A, B, S\}$
  - $\Sigma = \{a, b\}$
  - $R = \{S \rightarrow A, S \rightarrow abA, S \rightarrow aB, A \rightarrow a, B \rightarrow Sb\}$
- Solution:
  - $S \Rightarrow_G A \Rightarrow_G a$
  - $S \Rightarrow_G aB \Rightarrow_G aSb \Rightarrow_G aAb \Rightarrow_G aab$
  - $S \Rightarrow_G abA \Rightarrow_G aba$
  - $S \Rightarrow_G aB \Rightarrow_G aSb \Rightarrow_G aabAb \Rightarrow_G aabab$

CFG: Example 3

- Which words can be derived at most in 4 steps with G = {V, Σ, R, S} grammar from S?
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - R = {S → ABS, S → AB, A → aA, B → bB, S → *e*, A → a, B → b}
- Solution:
  - $S \Rightarrow_G ABS \Rightarrow_G aBS \Rightarrow_G abS \Rightarrow_G ab$
  - $S \Rightarrow_G AB \Rightarrow_G aB \Rightarrow_G ab$
  - $S \Rightarrow_G AB \Rightarrow_G aAB \Rightarrow_G aaB \Rightarrow_G aab$

CFG: Example 4

- Which words can be derived at most in 4 steps with $G = \{V, \Sigma, R, S\}$ grammar from S?
  - $V = \{a, b, A, B, S\}$
  - $\Sigma = \{a, b\}$
  - $R = \{S \rightarrow aSB,\ S \rightarrow bSA,\ S \rightarrow a,\ S \rightarrow b,\ A \rightarrow aS,\ B \rightarrow bS\}$

CFG: Example 4

- $R = \{S \rightarrow aSB, S \rightarrow bSA, S \rightarrow a, S \rightarrow b, A \rightarrow aS, B \rightarrow bS\}$
- Solution:
  - $S \Rightarrow_G aSB \Rightarrow_G aaB \Rightarrow_G aabS \Rightarrow_G aaba$
  - $S \Rightarrow_G bSA \Rightarrow_G baA \Rightarrow_G baaS \Rightarrow_G baaa$
  - $S \Rightarrow_G a$
  - $S \Rightarrow_G b$
  - $S \Rightarrow_G aSB \Rightarrow_G aSbS \Rightarrow_G aabS \Rightarrow_G aaba$

CFG: Example 5

- Give 3 different deductions to the word "abab"!
  - $V = \{a, b, A, B, S\}$
  - $\Sigma = \{a, b\}$
  - $R = \{S \to aB,\ S \to Ab,\ B \to bS,\ A \to Sa,\ S \to e\}$
- Solution:
  - $S \Rightarrow_G aB \Rightarrow_G abS \Rightarrow_G abaB \Rightarrow_G ababS \Rightarrow_G abab$
  - $S \Rightarrow_G aB \Rightarrow_G abS \Rightarrow_G abAb \Rightarrow_G abSab \Rightarrow_G abab$
  - $S \Rightarrow_G Ab \Rightarrow_G Sab \Rightarrow_G Abab \Rightarrow_G Sabab \Rightarrow_G abab$

CFG: Example 6

- Give 3 different deductions to the word "abba"!
  - $V = \{a, b, S\}$
  - $\Sigma = \{a, b\}$
  - $R = \{S \to SS,\ S \to abS,\ S \to Sba,\ S \to e\}$
- Solution:
  - $S \Rightarrow_G abS \Rightarrow_G abSba \Rightarrow_G abba$
  - $S \Rightarrow_G SS \Rightarrow_G abSS \Rightarrow_G abSSba \Rightarrow_G abSba \Rightarrow_G abba$
  - $S \Rightarrow_G Sba \Rightarrow_G abSba \Rightarrow_G abba$

CFG: Example 7

- Give context-free grammar G such that L(G) = L!
  - L = {w $\in$ {a, b }* | |w| is divisible with 3}
- Solution 1:
  - S $\rightarrow$ AAAS | *e*
  - A $\rightarrow$ a | b
- Solution 2:
  - S $\rightarrow$ aaaS | aabS | abbS | bbbS | bbaS | baaS | babS | abaS | *e*

CFG: Example 8

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a\}$
  - $L = \{a^n \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $V = \{S\}$
  - $R = \{(S, aS), (S, e)\}$
    - $S \rightarrow aS \mid e$

  - e.g.: S => aS => aaS => aa

CFG: Example 9

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^n \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $V = \{S\}$
  - $R = \{(S, aSb), (S, e)\}$
    - $S \rightarrow aSb \mid e$

  - e.g.: S => aSb => aaSbb => aabb

CFG: Example 10

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c\}$
  - $L = \{a^n c b^n \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $V = \{S\}$
  - $R = \{(S, aSb), (S, c)\}$
    - $S \rightarrow aSb \mid c$

  - e.g.: S => aSb => aaSbb => aacbb

CFG: Example 11

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c\}$
  - $L = \{a^n b^n c \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $V = \{S, Q\}$
  - $R = \{(S, Qc), (Q, aQb), (Q, e)\}$
    - $S \rightarrow Qc$
    - $Q \rightarrow aQb \mid e$

$$\underbrace{a^n b^n c}_{S}$$

$$\underbrace{\quad S \quad}_{Q}$$

  - e.g.: S => Qc => aQbc => aaQbbc => aabbc

CFG: Example 12

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^n a^m b^m \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow PQ$
  - $Q \rightarrow aQb \mid e$
  - $P \rightarrow aPb \mid e$

$$\underbrace{a^n b^n a^m b^m}_{S}$$

$$S = \underbrace{}_{P}\underbrace{}_{Q}$$

  - e.g.: S => PQ => aQbP => abP => abaPb => abaaPbb => abaabb

CFG: Example 13

- Give CFG G such that L(G) = L!
  - Σ = {a, b, c}
  - L = $\{a^n b^n c a^m b^m \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow PcQ$
  - $Q \rightarrow aQb \mid e$
  - $P \rightarrow aPb \mid e$

$$\underbrace{\underbrace{a^n b^n}_{P}\, c\, \underbrace{a^m b^m}_{Q}}_{S}$$

  - e.g.: S => PcQ => aPbcQ => aaPbbcQ => aabbcQ => aabbc

CFG: Example 14

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c\}$
  - $L = \{a^n b^n a^m c b^m \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow PQ$
  - $Q \rightarrow aQb \mid c$
  - $P \rightarrow aPb \mid e$

  - e.g.: S => PQ => PQ => PaQb => Pacb => aPbacb => abacb

CFG: Example 15

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^{n+1}b^n \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $L = \{a^n a b^n \in \Sigma^* \mid n \geq 0\}$
  - $S \rightarrow aSb \mid a$

  - e.g.: S => aSb => aaSbb => aaabb

CFG: Example 16

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^{3n} \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $S \rightarrow aSbbb \mid e$
  - e.g.: S => aSbbb => aaSbbbbbb => aabbbbbb

CFG: Example 17

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^{2n}b^{3n}a^{m+1}b^{2m+2} \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow PQ$
  - $P \rightarrow aaPbbb \mid e$
  - $Q \rightarrow aQbb \mid abb$

  - e.g.: S => PQ => aaPbbbQ => aabbbQ => aabbbabb

$$\underbrace{\underbrace{a^{2n}b^{3n}}_{P}\,\underbrace{a^{m+1}b^{2m+2}}_{Q}}_{S}$$

CFG: Example 18

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^m \in \Sigma^* \mid n, m \geq 0 \text{ and } n > m\}$
- Solution:
  - $L = \{a^* a^n b^n \in \Sigma^* \mid n \geq 0\}$
  - $S \rightarrow AB$
  - $A \rightarrow aA \mid a$
  - $B \rightarrow aBb \mid e$

  - e.g.: S => AB => aAB => aaAB => aaaAB => aaaB => aaaaBb => aaaab

CFG: Example 19

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^m \in \Sigma^* \mid n, m \geq 0 \text{ and } n < m\}$
- Solution:
  - $L = \{a^n b^n b^* \in \Sigma^* \mid n \geq 0\}$
  - $S \rightarrow BA$
  - $A \rightarrow bA \mid b$
  - $B \rightarrow aBb \mid e$

  - e.g.: S => BA => aBbA = >abA => abbA => abbb

CFG: Example 20

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^m \in \Sigma^* \mid n, m \geq 0 \text{ and } n \neq m\}$
- Solution:
  - $L = \{a^n b^m \in \Sigma^* \mid n < m \text{ or } n > m \}$
  - $S \rightarrow CD \mid DE$
  - $C \rightarrow aC \mid a$
  - $D \rightarrow aDb \mid e$
  - $E \rightarrow bE \mid b$

  - e.g.: S => CD => aCD => aaD => aaaDb => aaab

CFG: Example 21

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c, d\}$
  - $L = \{a^n c^m d^m b^n \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow aSb \mid A$
  - $A \rightarrow cAd \mid e$

$$\underbrace{a^n \underbrace{c^m d^m}_{A} b^n}_{S}$$

  - e.g.: S => aSb => aaSbb => aaAbb => aacAdbb => aacdbb

CFG: Example 22

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{(ab)^{2n+1}b^m a^{m+3} b^n \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $L = \{((ab)^2)^n ab[b^m a^m aaa]b^n \in \Sigma^* \mid n, m \geq 0\}$
  - $S \to (ab)^2Sb \mid abA$
  - $A \to bAa \mid aaa$

  - e.g.: S => abA => abbAa => abbaaaa

CFG: Example 23

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{(ab)^{2n+1}b^m a^{m+3} a^p b^{p+1} b^n \in \Sigma^* \mid n, m, p \geq 0\}$
- Solution:
  - $L = \{(ab)^{2n}ab[b^m aaaa^m][a^p bb^p]b^n \in \Sigma^* \mid n, m, p \geq 0\}$
  - $S \rightarrow ababSb \mid abAP$
  - $A \rightarrow bAa \mid aaa$
  - $P \rightarrow aPb \mid b$

  - e.g.: S => ababSb => abababAPb => abababaaaPb => abababaaabb

CFG: Example 24

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, h, i\}$
  - $L = \{a^n h^k b^{2m} a h a^{m+1} b^n i^p \in \Sigma^* \mid n, m, k, p \geq 0\}$
- Solution:
  - $S \rightarrow JP$
  - $J \rightarrow aJb \mid KM$
  - $K \rightarrow hK \mid e$
  - $M \rightarrow bbMa \mid aha$
  - $P \rightarrow iP \mid e$

  - e.g.: S => JP => KMP => hKMP => hMP => hbbMaP => hbbahaaP => hbbahaaiP => hbbahaai

CFG: Example 25

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c\}$
  - $L = \{(ac)^n(bc)^m b^n \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow N$
    - N the non-terminal corresponds to superscript n
  - $N \rightarrow acNb \mid M$
  - $M \rightarrow bcM \mid e$

  - e.g.: S => N => acNb => acMb => acbcMb => acbcb

CFG: Example 26

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c, k\}$
  - $L = \{a^k(ac)^n b^m c^m b^n k^k \in \Sigma^* \mid n, m, k \geq 0\}$
- Solution:
  - $S \rightarrow K$
  - $K \rightarrow aKk \mid N$
  - $N \rightarrow acNb \mid M$
  - $M \rightarrow bMc \mid e$

  - e.g.: S => K => aKk => aNk => aacNbk => aacMbk => aacbMcbk => aacbcbk

CFG: Example 27

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^{n+1}b^{n+3} \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $L = \{a^n abbbb^n \in \Sigma^* \mid n \geq 0\}$
  - $S \rightarrow aSb \mid abbb$

  - e.g.: S => aSb => aabbbb

CFG: Example 28

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{a^n b^m a^n \in \Sigma^* \mid n, m \geq 0\}$
- Solution:
  - $S \rightarrow N$
  - $N \rightarrow aNa \mid M$
  - $M \rightarrow bM \mid e$

  - e.g.: S => N => aNa => aaNaa => aaMaa => aabMaa => aabbMaa => aabbaa

CFG: Example 29

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c\}$
  - $L = \{a^n c b^n c \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $S \rightarrow Nc$
    - ending c derived
  - $N \rightarrow aNb \mid e$

CFG: Example 30

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{b^n a^n \in \Sigma^* \mid n \geq 0\} \cup \{a^{3n} b^n \in \Sigma^* \mid n \geq 0\}$
- Solution:
  - $S \rightarrow N1 \mid N2$
  - $N1 \rightarrow bN1a \mid e$
  - $N2 \rightarrow aaaN2b \mid e$

CFG: Example 31

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c, d\}$
  - $L = \{a^m b^n c^p d^r \in \Sigma^* \mid m+n = p+r, m, n, p, r \geq 0\}$
- Solution:
  - $S \rightarrow A$
  - $A \rightarrow aAd \mid B \mid C$
    - always one symbol is created from {a, b} and {c, d}
  - $B \rightarrow aBc \mid D$
  - $C \rightarrow bCd \mid D$
  - $D \rightarrow bDc \mid e$
  - e.g.: S => A => B => aBc => aaBcc => aaDcc => aabDccc => aabccc

CFG: Example 32

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c, d\}$
  - $L = \{a^{n+2}bc^{2m}d^{p}a^{n+3}c^{2q}(ad)^{m} \in \Sigma^* \mid n, m, p, q \geq 0\}$
- Solution:
  - L is not context free because it is a modification of $\{a^{n}b^{m}c^{n}d^{m} \in \Sigma^* \mid n, m \geq 0\}$ which is not context free
    - the superscripts can be regarded as different kinds of opening and closing parenthesis
    - here the are parenthesis neither nested ([]) nor next to each other ()[] but across ([)]

CFG: Example 32

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b, c, d\}$
  - $L = \{a^m b^m d^{n+2} b c^4 d^{2p} a^{n+3} c^{2q} (ad)^{n+3} \in \Sigma^* \mid n, m, p, q \geq 0\}$
- Solution:
  - L is not context free because it is a modification of $\{a^n b^n c^n \in \Sigma^* \mid n \geq 0\}$ which is not context free

CFG: Example 33

- Give CFG G such that L(G) = L!
  - $\Sigma = \{a, b\}$
  - $L = \{w \in \Sigma^* \mid \#a \neq \#b\}$

CFG: Example 33

- Solution:
  - $S \rightarrow U \mid V$
  - $U \rightarrow TaU \mid TaT$
  - $V \rightarrow TbV \mid TbT$
  - $T \rightarrow aTbT \mid bTaT \mid e$
    - T: strings with the same number of a's as b's
    - U: strings with more a's than b's
    - V: strings with more b's than a's

  - e.g.: S => U => TaU => bTaTaU => baaU => baaTaT => baaa

CFG: Example 34

- Give CFG G such that L(G) = L!
  - $L = \{b^n a^m b^{2n} : n \geq 0, m > 0\}$
  - beware: > and not ≥

CFG: Example 34

- Solution:
  - $L = \{b^n a a^m b^{2n} : n \geq 0, m \geq 0\}$
  - $S \rightarrow bSbb \mid aA$
  - $A \rightarrow aA \mid e$

  - e.g.: S => bSbb => bAbb => baAbb => baaAbb => baabb

RG $\rightarrow$ CFG: Example 1

- Give CFG G such that L(G) = L(R)!
  - R = $((a \cup b)a)^*$

RG → CFG: Example 1

- Solution 1:
  - S → SXa | *e*
  - X → a | b
- Solution 2:
  - S → *e* | YS (Kleen star)
  - Y → XA (concatenation)
  - X → a | b (union)

RG $\rightarrow$ CFG: Example 2

- Give CFG G such that L(G) = L(R)!
  - $R = (ab)^* \cup a$
- Solution:
  - $S \rightarrow A \mid a$
  - $A \rightarrow abA \mid e$

  - e.g.: S => a

RG $\rightarrow$ CFG: Example 3

- Give CFG G such that L(G) = L(R)!
  - $R = (ba \cup b)^*a$

RG → CFG: Example 3

- R = (ba ∪ b)*a
- Solution: RE → CFG
  - S → XY (concatenation)
  - Y → a
  - X → ZX | *e* (Kleen start)
  - Z → E | F (union)
  - E → ba
  - F → b

  - e.g.: S =>XY => ZXY => EXY => baXY => baY => baa

RG $\rightarrow$ CFG: Example 4

- Give CFG G such that L(G) = L(R)!
  - R = $a^*(ba \cup b)$
- Solution:
  - $S \rightarrow Aba \mid Ab$
  - $A \rightarrow aA \mid e$

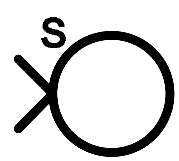
  - e.g.: S => Ab => aAb => aaAb => aab

RG → NFA: Example 1

- Construct NFA M such that L(M) = L(G)!
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, B → baS}

RG → NFA: Example 1

- Step a1: introduce a state for non-terminal S (this will be the initial state because S is the starting non-terminal)
  - V = {a, b, A, B, **S**}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, B → baS}

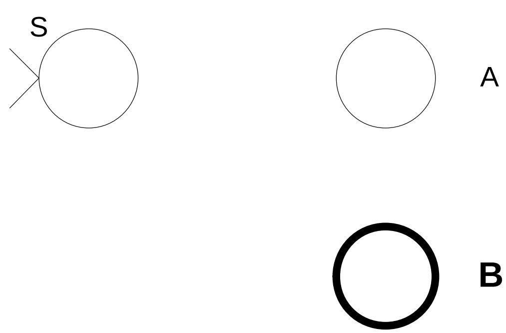


RG → NFA: Example 1

- Step a2: introduce a state for non-terminal A
  - V = {a, b, **A**, B, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, B → baS}

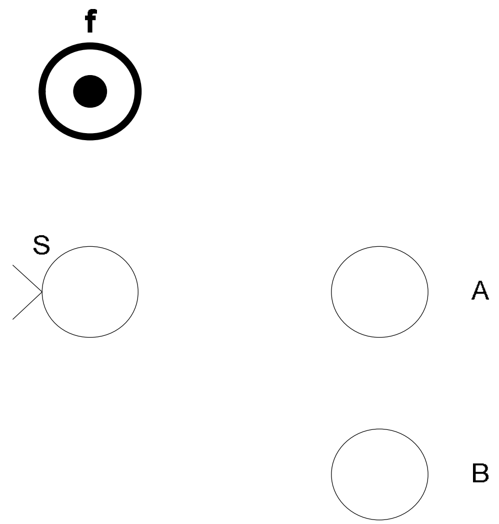
RG → NFA: Example 1

- Step a3: introduce a state for non-terminal B
  - V = {a, b, A, **B**, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, B → baS}



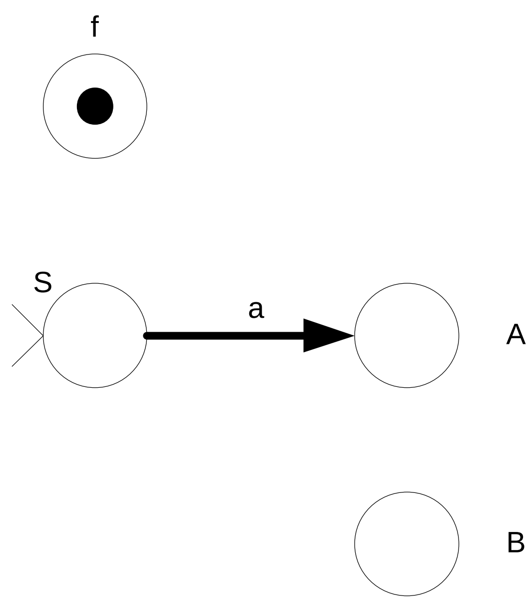
RG → NFA: Example 1

- Step b: introduce a state for non-terminal f (this will be the final state)
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, B → baS}



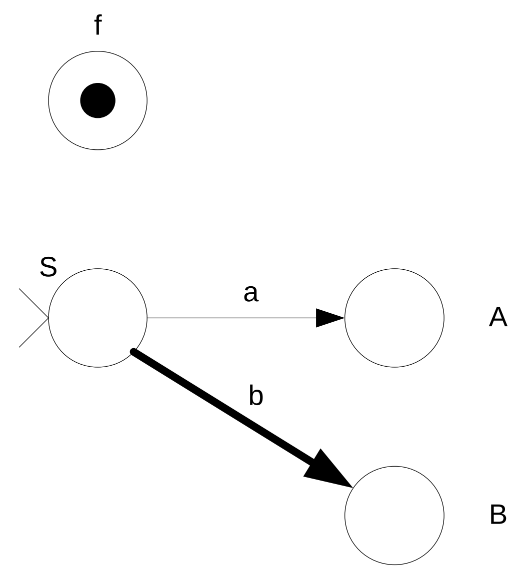
RG → NFA: Example 1

- Step c1: introduce a new arc for S → aA
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {**S** → **aA** | bB | *e*, A → abS, B → baS}



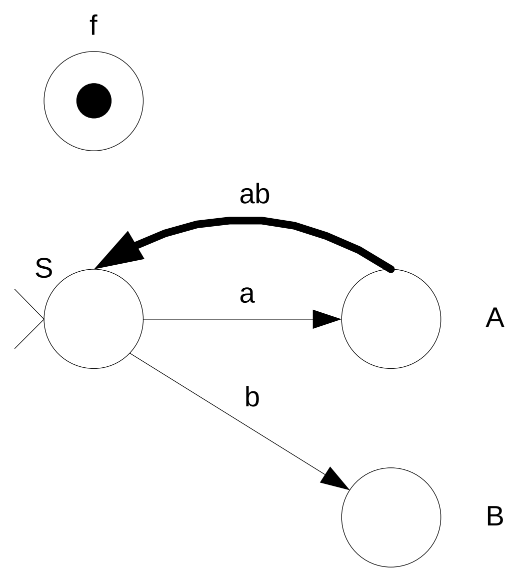
RG → NFA: Example 1

- Step c2: introduce a new arc for S → bB
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {**S** → aA | **bB** | *e*, A → abS, B → baS}



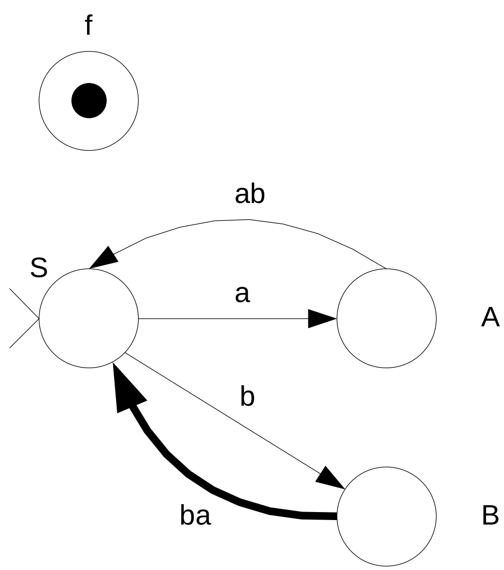
RG → NFA: Example 1

- Step c3: introduce a new arc for A → abS
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, **A → abS**, B → baS}



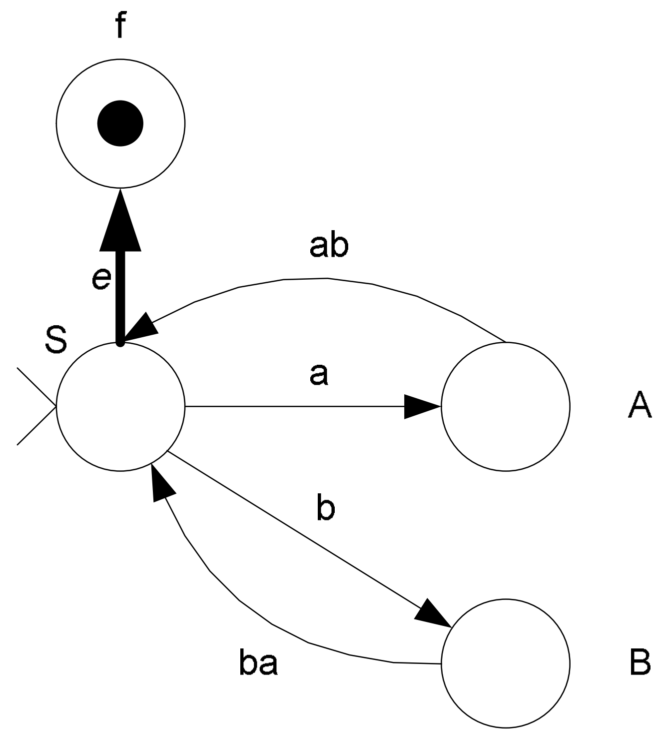
RG → NFA: Example 1

- Step c4: introduce a new arc for B → baS
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA | bB | *e*, A → abS, **B** → **baS**}



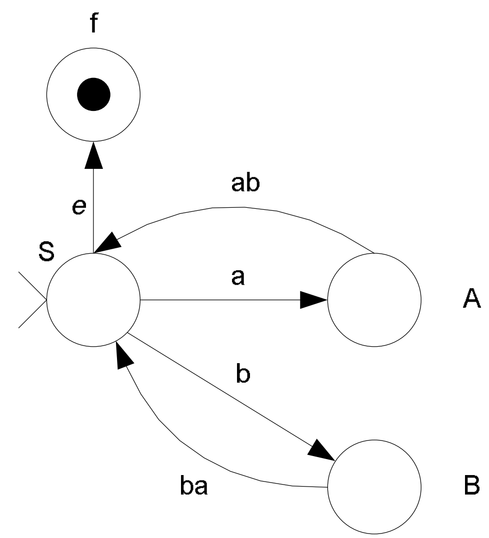
RG → NFA: Example 1

- Step d1: introduce a new arc for S → *e*
  - V = {a, b, A, B, S}
  - Σ = {a, b}
  - S = S
  - R = {**S** → aA | bB | ***e***,
    A → abS, B → baS}



RG → NFA: Example 1

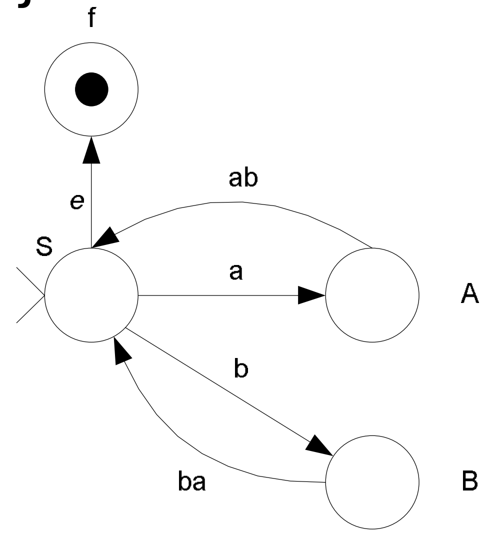
- R = {S → aA | bB | *e*, A → abS, B → baS}



- Operation for aab:
  - (S, aab) |- (A, ab) |- (S, *e*) |- (f, *e*)
  - S => aA => aabS => aab

RG → NFA: Example 1

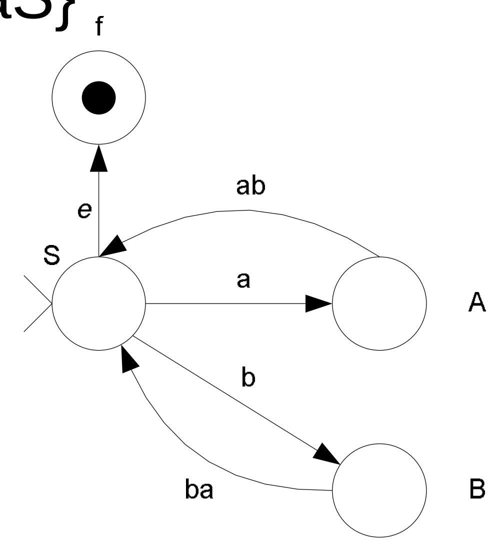
- R = {S → aA | bB | *e*, A → abS, B → baS}



- Operation for aabbbaaab:
  - (S, aabbbaaab) |- (A, abbbaaab) |- (S, bbaaab) |- (B, baaab) |- (S, aab) |- (A, ab) |- (S, *e*) |- (f, *e*)
  - S => aA => aabS => aabbB => aabbbaS => aabbbaaA => aabbbaaabS => aabbbaaab

RG → NFA: Example 1

- R = {S → aA | bB | *e*, A → abS, B → baS}



- Operation for aabbba:
  – (S, aabbba) |- (A, abbba) |- (S, bba) |- (B, ba) |- (S, *e*) |- (f, *e*)
  – S => aA => aabS => aabbB => aabbbaS => aabbba

RG $\rightarrow$ NFA: Example 2

- Construct NFA M such that L(M) = L(RG)! Describe M with sets!
  - $V = \{a, b, A, B, C, S\}$
  - $\Sigma = \{a, b\}$
  - $S = S$
  - $R = \{S \rightarrow aA \mid bB,\ A \rightarrow aA \mid aC \mid bb,\ B \rightarrow e,\ C \rightarrow a\}$

RG $\rightarrow$ NFA: Example 2

- Step a1: introduce a state for non-terminal S (this will be the initial state because S is the starting non-terminal)
  - G({a, b, A, B, C, **S**}, {a, b}, {S $\rightarrow$ aA | bB, A $\rightarrow$ aA | aC | bb, B $\rightarrow$ *e*, C $\rightarrow$ a}, S)
  - M({**S**})

RG $\rightarrow$ NFA: Example 2

- Step a2: introduce a state for non-terminal A
  - G({a, b, **A**, B, C, S}, {a, b}, {S $\rightarrow$ aA | bB, A $\rightarrow$ aA | aC | bb, B $\rightarrow$ *e*, C $\rightarrow$ a}, S)

  - M({S**, A**})

RG → NFA: Example 2

- Step a3: introduce a state for non-terminal B
  - G({a, b, A, **B**, C, S}, {a, b}, {S → aA | bB, A → aA | aC | bb, B → *e*, C → a}, S)
  - M({S, A, **B**})

RG $\rightarrow$ NFA: Example 2

- Step a4: introduce a state for non-terminal C
  - G({a, b, A, B, **C**, S}, {a, b}, {S $\rightarrow$ aA | bB, A $\rightarrow$ aA | aC | bb, B $\rightarrow$ *e*, C $\rightarrow$ a}, S)

  - M({S, A, B, **C**})

RG → NFA: Example 2

- Step b1: introduce the elements of Σ
  - G({a, b, A, B, C, S}, {**a,** b}, {S → aA | bB,
    A → aA | aC | bb, B → *e*, C → a}, S)

  - M({S, A, B, C}, {**a**})

RG → NFA: Example 2

- Step b2: introduce the elements of Σ
  - G({a, b, A, B, C, S}, {a, **b**}, {S → aA | bB, A → aA | aC | bb, B → *e*, C → a}, S)

  - M({S, A, B, C}, {a, **b**})

RG → NFA: Example 2

- Step c1: introduce a new transition for S → aA
  - G({a, b, A, B, C, S}, {a, b}, {**S → aA** | bB, A → aA | aC | bb, B → *e*, C → a}, S)
  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {**(S, a, A)**}

RG → NFA: Example 2

- Step c1: introduce a new transition for S → bB
  - G({a, b, A, B, C, S}, {a, b}, {**S** → aA | **bB**,
    A → aA | aC | bb, B → *e*, C → a}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), **(S, b, B)**}

RG → NFA: Example 2

- Step c1: introduce a new transition for A → aA
  - G({a, b, A, B, C, S}, {a, b}, {S → aA | bB,
    A → **aA** | aC | bb, B → *e*, C → a}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), (S, b, B), **(A, a, A)**}

RG $\rightarrow$ NFA: Example 2

- Step c1: introduce a new transition for A $\rightarrow$ aC
  - G({a, b, A, B, C, S}, {a, b}, {S $\rightarrow$ aA | bB, **A** $\rightarrow$ aA | **aC** | bb, B $\rightarrow$ *e*, C $\rightarrow$ a}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), (S, b, B), (A, a, A), **(A, a, C)**}

RG → NFA: Example 2

- Step d1: introduce a new transition for A → bb
  - G({a, b, A, B, C, S}, {a, b}, {S → aA | bB,
    A → aA | aC | **bb**, B → *e*, C → a}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), (S, b, B), (A, a, A), (A, a, C),
      (A, bb, f)}

RG $\rightarrow$ NFA: Example 2

- Step d2: introduce a new transition for B $\rightarrow$ e
  - G({a, b, A, B, C, S}, {a, b}, {S $\rightarrow$ aA | bB, A $\rightarrow$ aA | aC | bb, **B** $\rightarrow$ ***e***, C $\rightarrow$ a}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), (S, b, B), (A, a, A), (A, a, C), (A, bb, f), **(B, *e*, f)**}

RG → NFA: Example 2

- Step d3: introduce a new transition for C → a
  - G({a, b, A, B, C, S}, {a, b}, {S → aA | bB, A → aA | aC | bb, B → *e*, **C** → **a**}, S)

  - M({S, A, B, C}, {a, b}, **Δ**)
    - Δ = {(S, a, A), (S, b, B), (A, a, A), (A, a, C), (A, bb, f), (B, *e*, f), **(C, a, f)**}

RG $\rightarrow$ NFA: Example 2

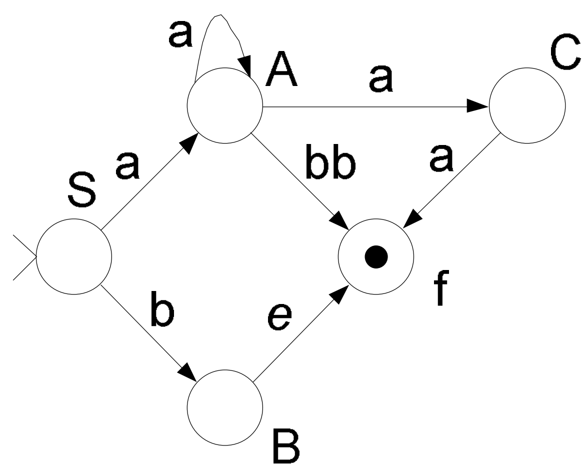
- Step: introduce the initial state
  - G({a, b, A, B, C, S}, {a, b}, {S $\rightarrow$ aA | bB, A $\rightarrow$ aA | aC | bb, B $\rightarrow$ *e*, C $\rightarrow$ a}, **S**)

  - M({S, A, B, C}, {a, b}, Δ, **S**)
    - Δ = {(S, a, A), (S, b, B), (A, a, A), (A, a, C), (A, bb, f), (B, *e*, f), (C, a, f)}

RG → NFA: Example 2

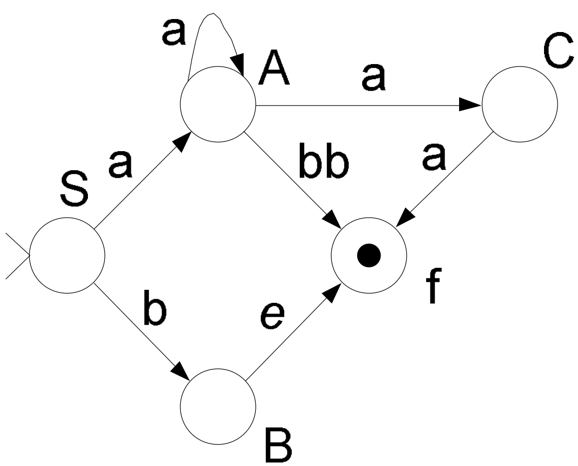
- Step: introduce the final state
  - G({a, b, A, B, C, S}, {a, b}, {S → aA | bB, A → aA | aC | bb, B → *e*, C → a}, S)

  - M({S, A, B, C}, {a, b}, Δ, S, {**f**})
    - Δ = {(S, a, A), (S, b, B), (A, a, A), (A, a, C), (A, bb, f), (B, *e*, f), (C, a, f)}



RG → NFA: Example 2

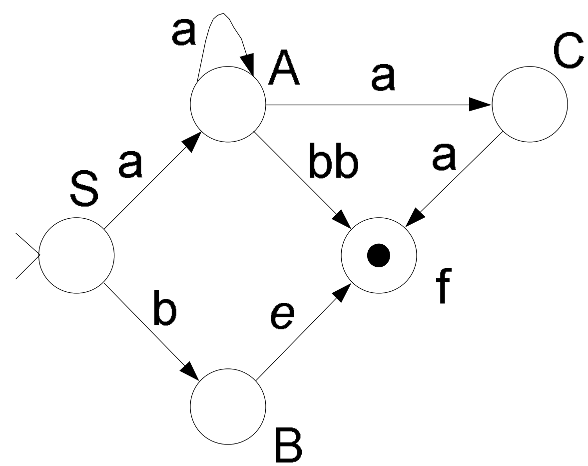
- R = {S → aA | bB, A → aA | aC | bb, B → *e*, C → a}



- Operation for aabb:
  - (S, aabb) |- (A, abb) |- (A, bb) |- (f, *e*)
  - S => aA => aaA => aabb

RG → NFA: Example 2

- R = {S → aA | bB, A → aA | aC | bb, B → *e*, C → a}



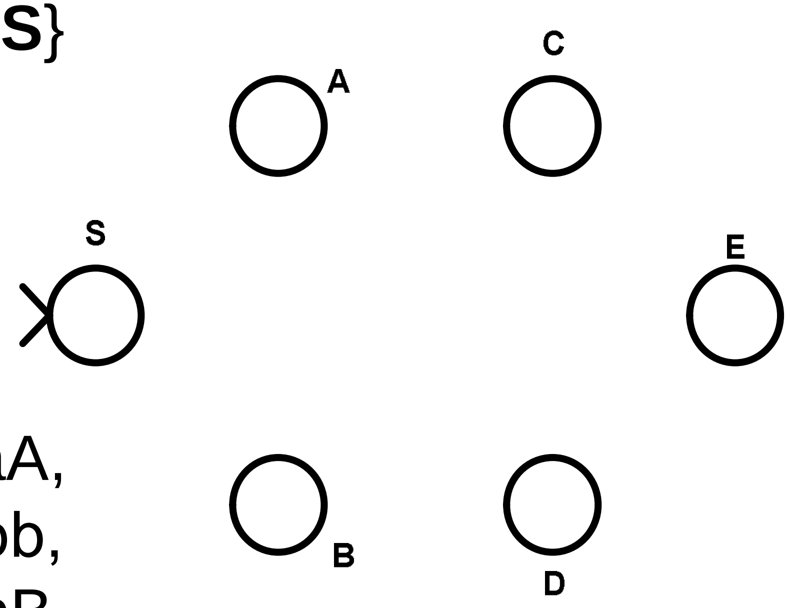
- Operation for aaaa:
  – (S, aaaa) |- (A, aaa) |- (A, aa) |- (C, a) |- (f, *e*)
  – S => aA => aaA => aaaC => aaaa

RG → NFA: Example 3

- Construct NFA M such that L(M) = L(G)!
  - V = {a, b, A, B, C, D, E, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA, S → aB, S → aC, S → bD, S → bA, S → bB, A → aA, A → bA, A → aC, A → bb, C → aE, B → aB, B → bB, B → bD, D → bE, D → ab, E → *e*}

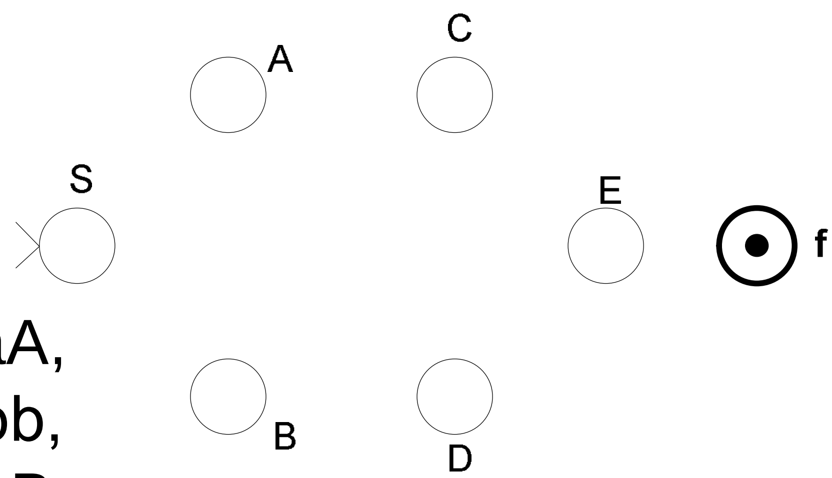
RG → NFA: Example 3

- Steps a: introduce states
  - V = {a, b, **A**, **B**, **C**, **D**, **E**, **S**}
  - Σ = {a, b}
  - S = S
  - R = {S → aA, S → aB,
    S → aC, S → bD,
    S → bA, S → bB, A → aA,
    A → bA, A → aC, A → bb,
    C → aE, B → aB, B → bB,
    B → bD, D → bE, D → ab, E → *e*}



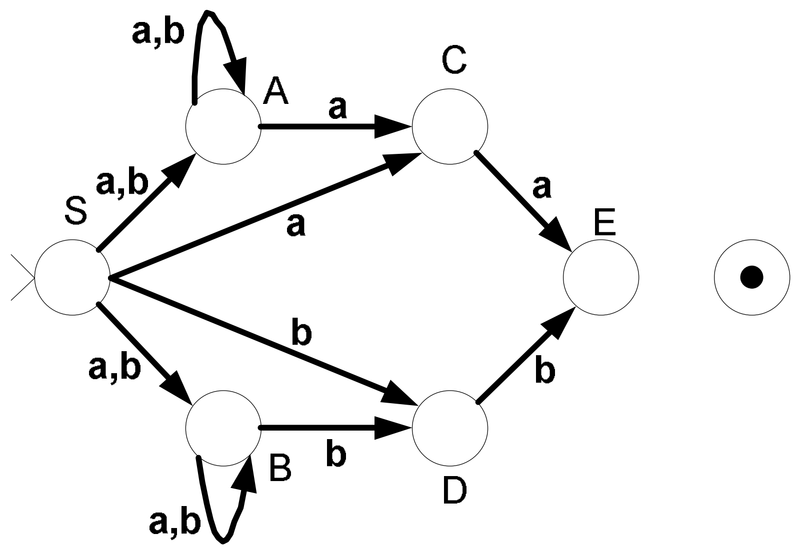
RG → NFA: Example 3

- Step b: introduce final state
  - V = {a, b, A, B, C, D, E, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA, S → aB,
    S → aC, S → bD,
    S → bA, S → bB, A → aA,
    A → bA, A → aC, A → bb,
    C → aE, B → aB, B → bB,
    B → bD, D → bE, D → ab, E → *e*}



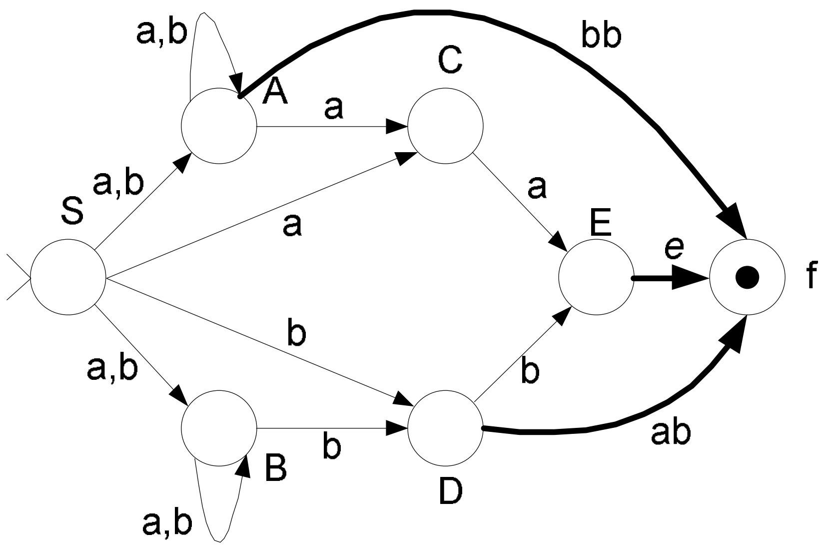
RG → NFA: Example 3

- Step c: introduce arcs
  - V = {a, b, A, B, C, D, E, S}
  - Σ = {a, b}
  - S = S
  - R = {**S → aA**, **S → aB**, **S → aC**, **S → bD**, **S → bA**, **S → bB**, **A → aA**, **A → bA**, **A → aC**, A → bb, **C → aE**, **B → aB**, **B → bB**, **B → bD**, **D → bE**, D → ab, E → *e*}



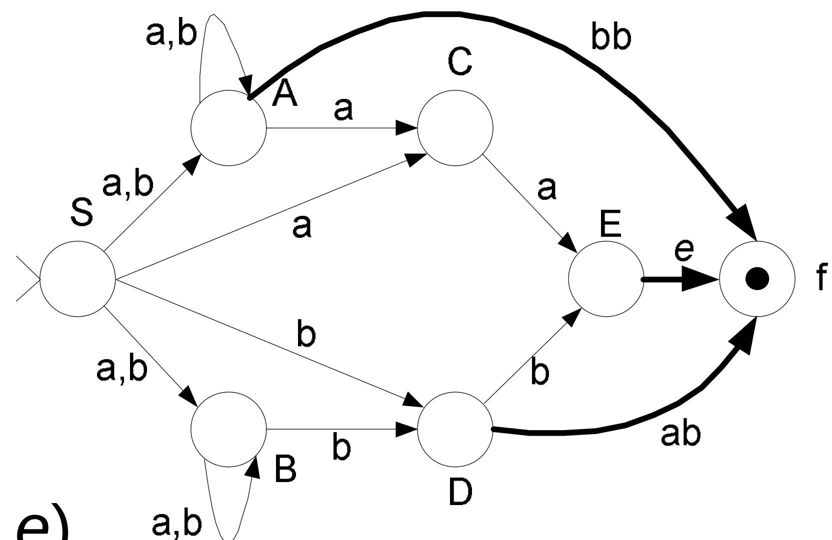
RG → NFA: Example 3

- Step d: introduce arcs
  - V = {a, b, A, B, C, D, E, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aA, S → aB,
    S → aC, S → bD,
    S → bA, S → bB, A → aA,
    A → bA, A → aC, **A** → **bb**,
    C → aE, B → aB, B → bB,
    B → bD, D → bE, **D** → **ab**, **E** → ***e***}



RG → NFA: Example 3

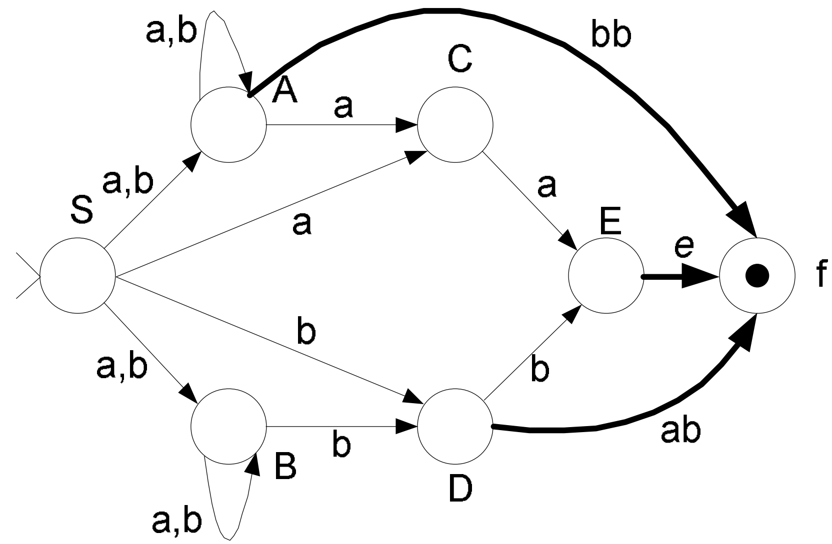
- R= {S → aA, S → aB, S → aC, S → bD, S → bA, S → bB, A → aA, A → bA, A → aC, A → bb, C → aE, B → aB, B → bB, B → bD, D → bE, D → ab, E → *e*}



- Operation for bb:
  - (S, bb) |- (D, b) |- (E, *e*) |- (f, *e*)
  - S => bD => bbE => bb

RG $\rightarrow$ NFA: Example 3

- R= {S $\rightarrow$ aA, S $\rightarrow$ aB, S $\rightarrow$ aC, S $\rightarrow$ bD, S $\rightarrow$ bA, S $\rightarrow$ bB, A $\rightarrow$ aA, A $\rightarrow$ bA, A $\rightarrow$ aC, A $\rightarrow$ bb, C $\rightarrow$ aE, B $\rightarrow$ aB, B $\rightarrow$ bB, B $\rightarrow$ bD, D $\rightarrow$ bE, D $\rightarrow$ ab, E $\rightarrow$ *e*}



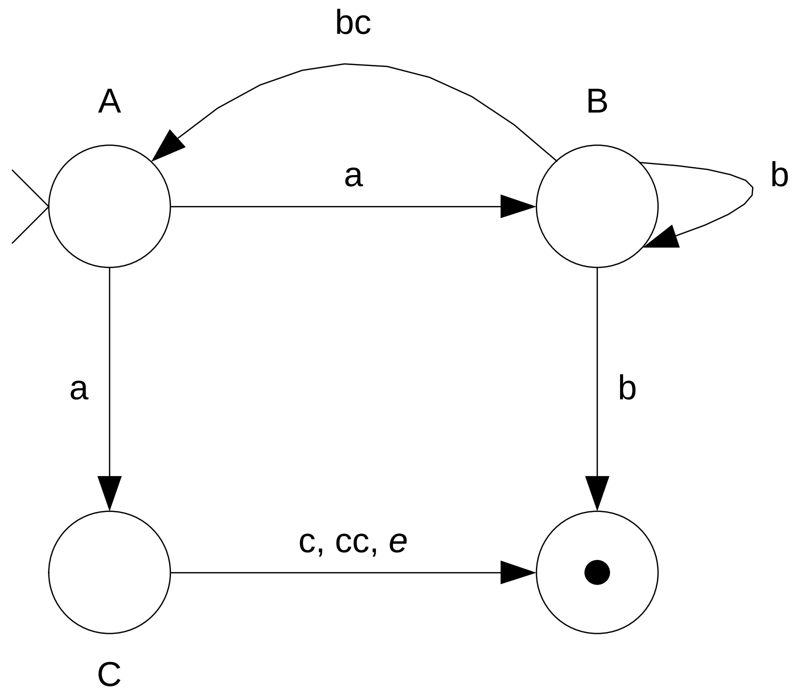
- Operation for abaa:
  - (S, abaa) |- (A, baa) |- (A, aa) |- (C, a) |- (E, *e*) |- (f, *e*)
  - S => aA => abA => abaC => abaaE => abaa

RG $\rightarrow$ NFA: Example 4

- Construct NFA M such that L(M) = L(G)!
  - V = {a, b, c, A, B, C}
  - Σ = {a, b, c}
  - S = A
  - R = {A $\rightarrow$ aB | aC, B $\rightarrow$ bcA | bB | b, C $\rightarrow$ c | cc | *e*}

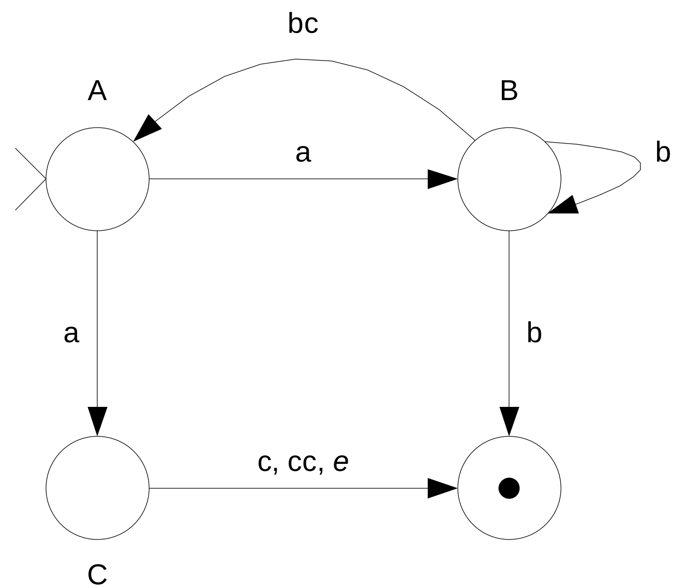
RG → NFA: Example 4

- R = {A → aB | aC, B → bcA | bB | b, C → c | cc | *e*}



RG → NFA: Example 4

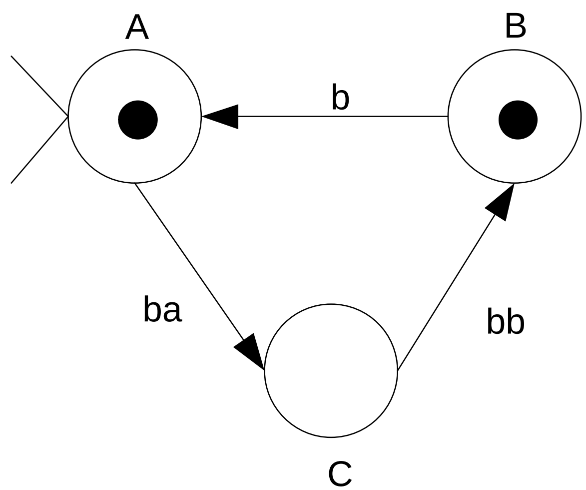
- R = {A → aB | aC, B → bcA | bB | b, C → c | cc | *e*}



- Operation for abaa:
  - (A, abcabb) |- (B, bcabb) |- (A, abb) |- (B, bb) |- (B, b) |- (f, *e*)
  - A => aB => abcA => abcaB => abcabB => abcabb

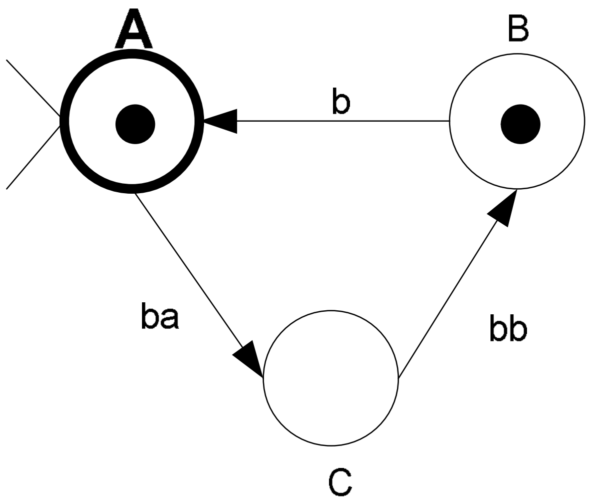
NFA → RG: Example 1

- Construct RG G such that L(G) = L(M)!



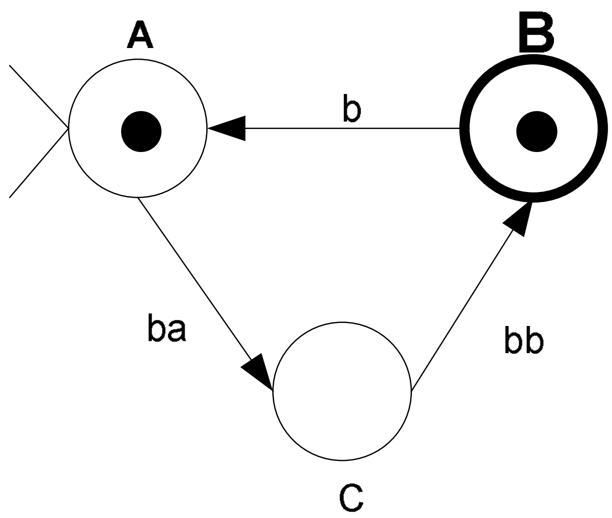
NFA → RG: Example 1

- Step a1: introduce non-terminals
  - V = {**A**}



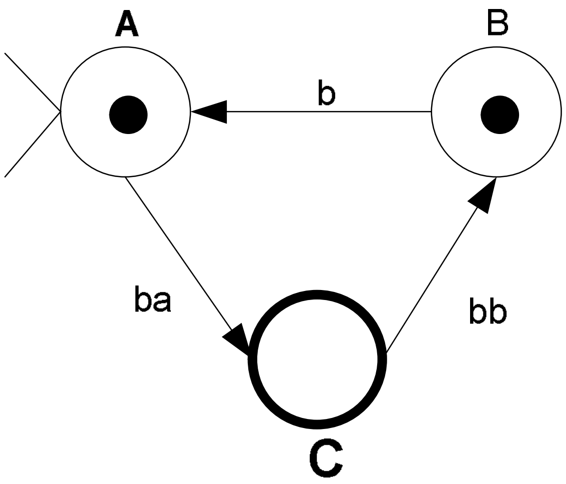
NFA → RG: Example 1

- Step a2: introduce non-terminals
  - V = {A, **B**}



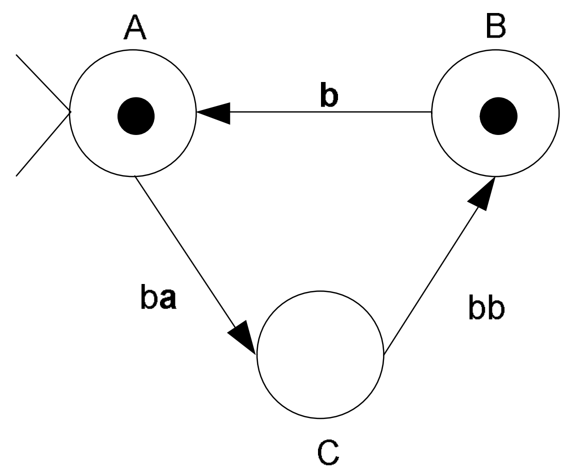
NFA → RG: Example 1

- Step a3: introduce non-terminals
  - V = {A, B, **C**}



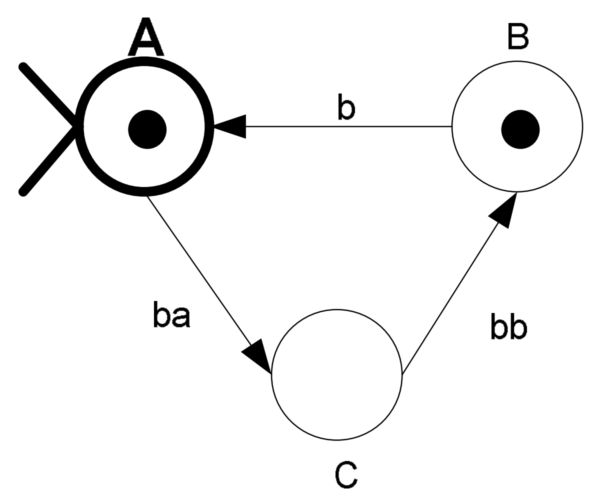
NFA → RG: Example 1

- Step: introduce Σ
  - V = {A, B, C}
  - **Σ = {a, b}**



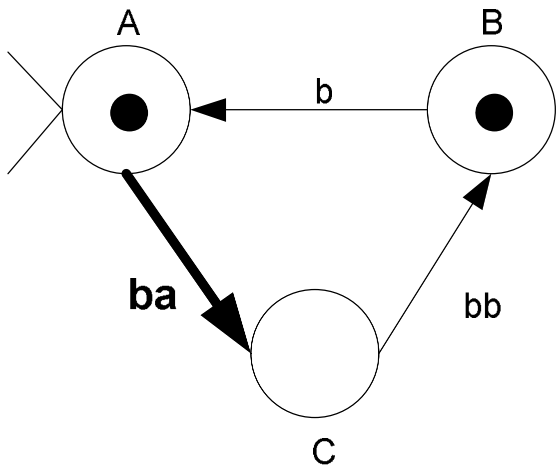
NFA → RG: Example 1

- Step: introduce the starting non-terminal
  - V = {A, B, C}
  - Σ = {a, b}
  - S = **A**



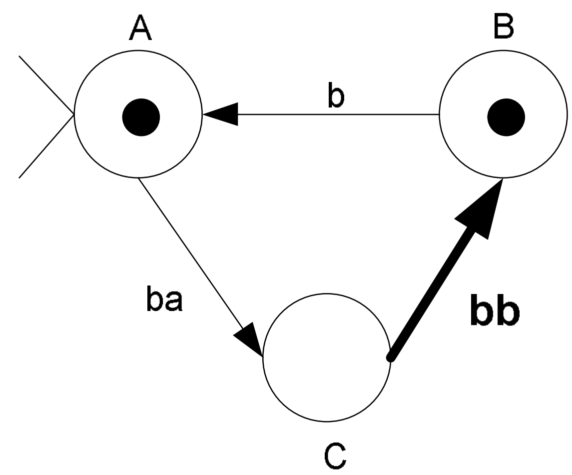
NFA → RG: Example 1

- Step b1: introduce rules for arcs
  - V = {A, B, C}
  - Σ = {a, b}
  - S = A
  - R = {**A** → **baC**}



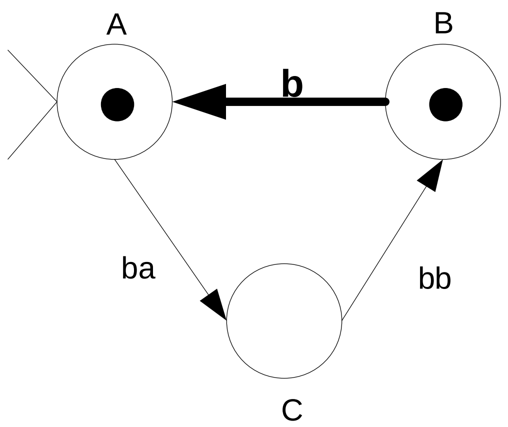
NFA → RG: Example 1

- Step b2: introduce rules for arcs
  - V = {A, B, C}
  - Σ = {a, b}
  - S = A
  - R = {A → baC, **C → bbB**}



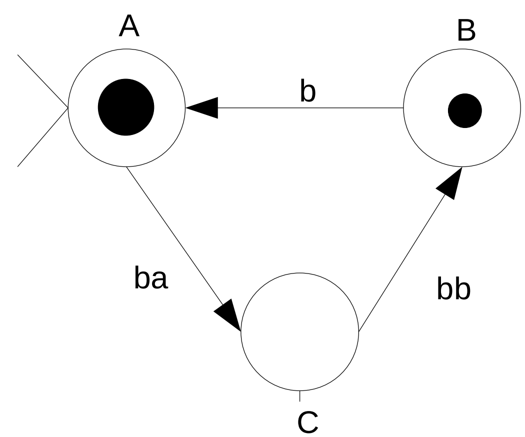
NFA → RG: Example 1

- Step b3: introduce rules for arcs
  - V = {A, B, C}
  - Σ = {a, b}
  - S = A
  - R = {A → baC, C → bbB, **B** → **bA**}



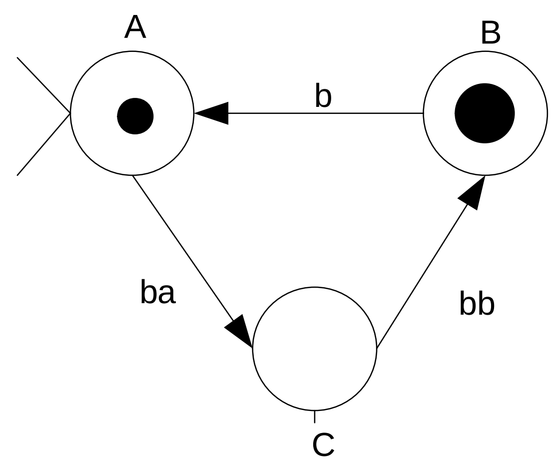
NFA → RG: Example 1

- Step c1: introduce rules for final states
  - V = {A, B, C}
  - Σ = {a, b}
  - S = A
  - R = {A → baC, C → bbB, B → bA, **A** → ***e***}



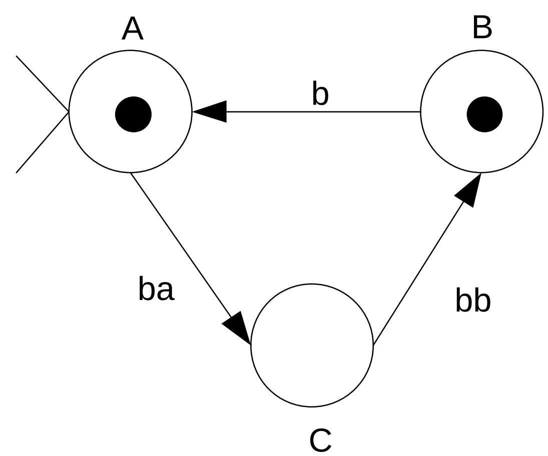
NFA → RG: Example 1

- Step c2: introduce rules for final states
  - V = {A, B, C}
  - Σ = {a, b}
  - S = A
  - R = {A → baC, C → bbB, B → bA,
    A → *e*, **B** → ***e***}



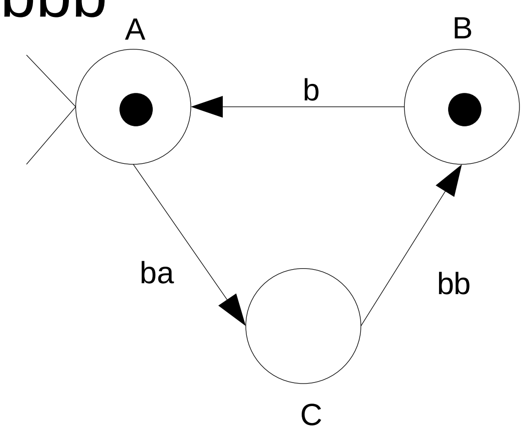
NFA → RG: Example 1

- $R = \{A \rightarrow baC,\ C \rightarrow bbB,\ B \rightarrow bA,\ A \rightarrow e,\ B \rightarrow e\}$
- Operation for babb:
  - (A, babb) |- (C, bb) |- (B, *e*)
  - A => baC => babbB => babb



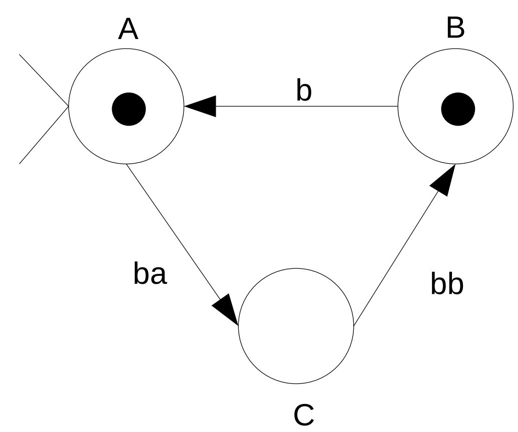
NFA → RG: Example 1

- R = {A → baC, C → bbB, B → bA, A → *e*, B → *e*}
- Operation for babbb:
  - (A, babbb) |- (C, bbb) |- (B, b) |- (A, *e*)
  - A => baC => babbB => babbbA => babbb



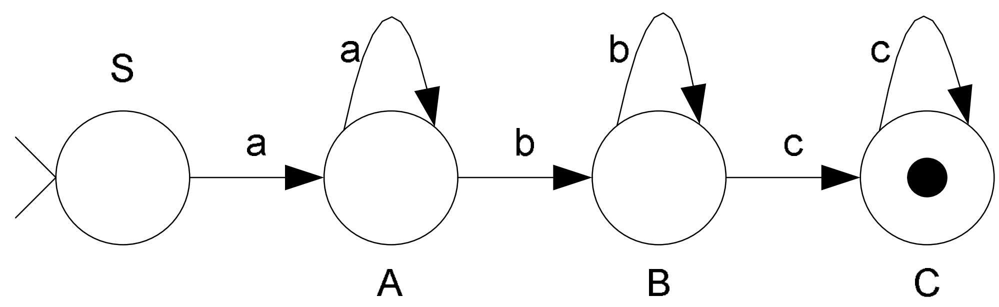
NFA → RG: Example 1

- R = {A → baC, C → bbB, B → bA, A → *e*, B → *e*}
- Operation for babbbbabb:
  - (A, babbbbabb) |- (C, bbbbabb) |- (B, bbabb) |- (A, babb) |- (C, bb) |- (B, *e*)
  - A => baC => babbB => babbbA => babbbbaC => babbbbabbB => babbbbabb



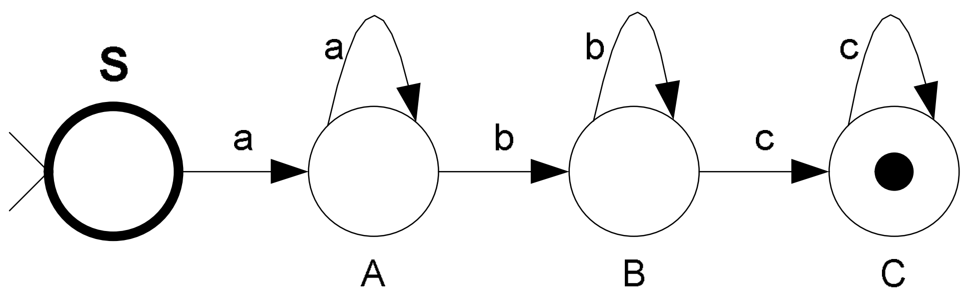
NFA → RG: Example 2

- Construct RG G such that L(G) = L(M)!



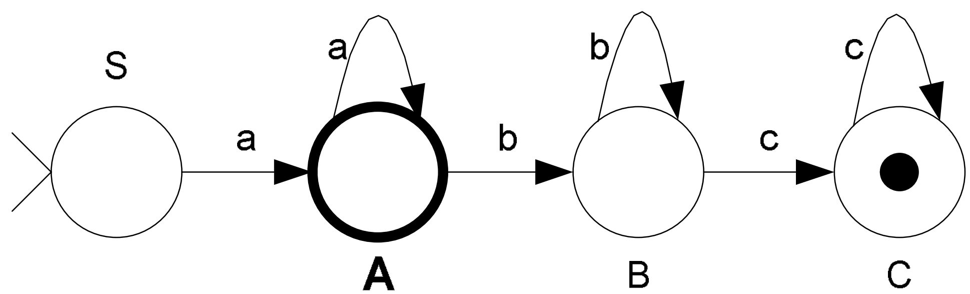
NFA → RG: Example 2

- Step a1: introduce non-terminals
  - V = {**S**}



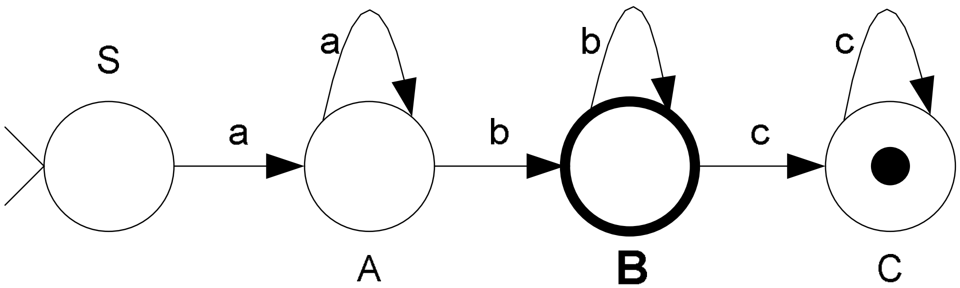
NFA → RG: Example 2

- Step a2: introduce non-terminals
  - V = {S, **A**}



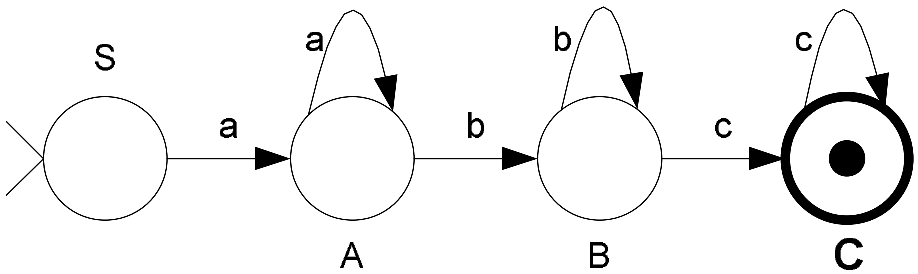
NFA → RG: Example 2

- Step a3: introduce non-terminals
  - V = {S, A, **B**}



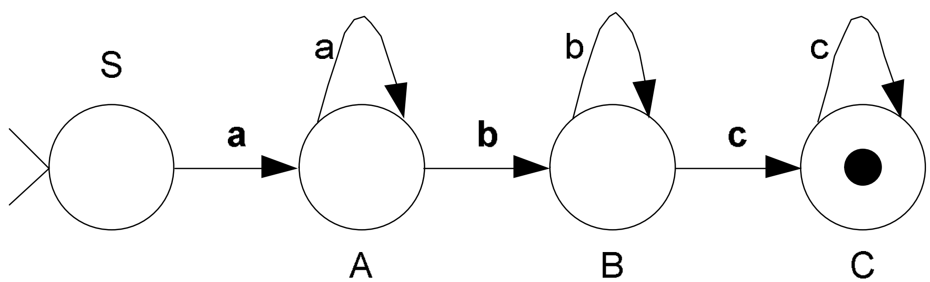
NFA → RG: Example 2

- Step a4: introduce non-terminals
  - V = {S, A, B, **C**}



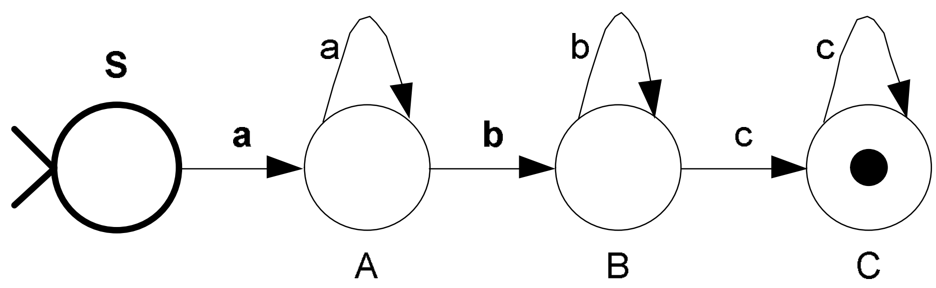
NFA → RG: Example 2

- Step: introduce Σ
  - V = {S, A, B, C}
  - Σ = {**a**, **b**, **c**}



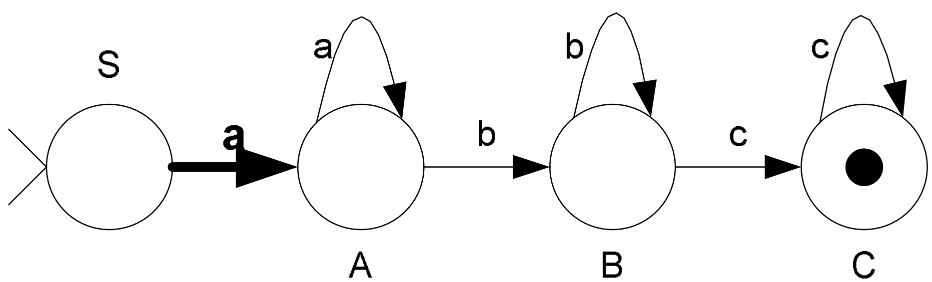
NFA → RG: Example 2

- Step: introduce the starting non-terminal
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = **S**



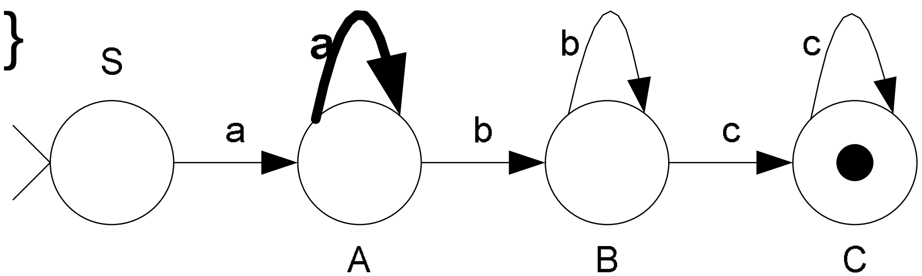
NFA → RG: Example 2

- Step b1: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {**(S, aA)**}



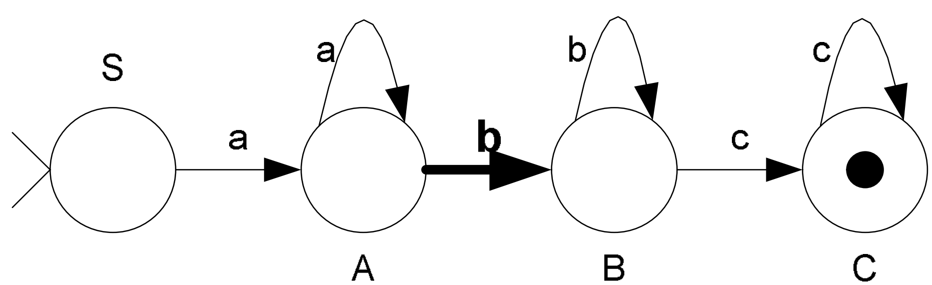
NFA → RG: Example 2

- Step b2: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), **(A, aA)**}



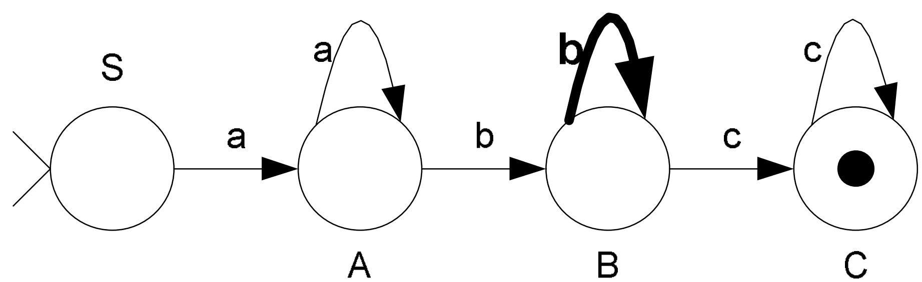
NFA → RG: Example 2

- Step b3: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), (A, aA), **(A, bB)**}



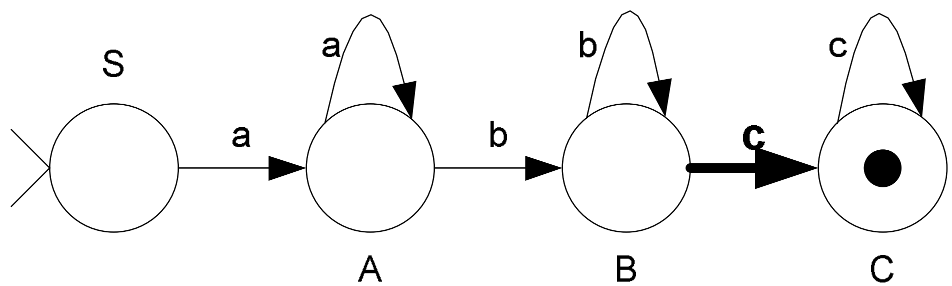
NFA → RG: Example 2

- Step b4: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), (A, aA), (A, bB), **(B, bB)**}



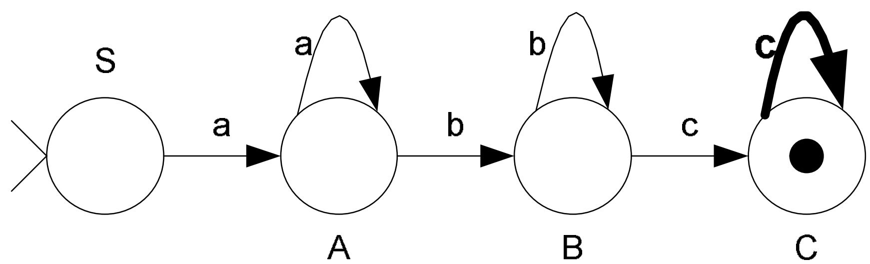
NFA → RG: Example 2

- Step b5: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), (A, aA), (A, bB), (B, bB), **(B, cC)**}



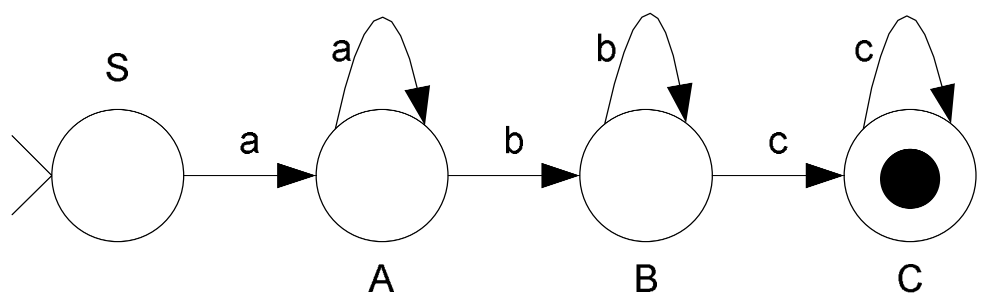
NFA → RG: Example 2

- Step b6: introduce rules for arcs
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), (A, aA), (A, bB), (B, bB), (B, cC), **(C, cC)**}



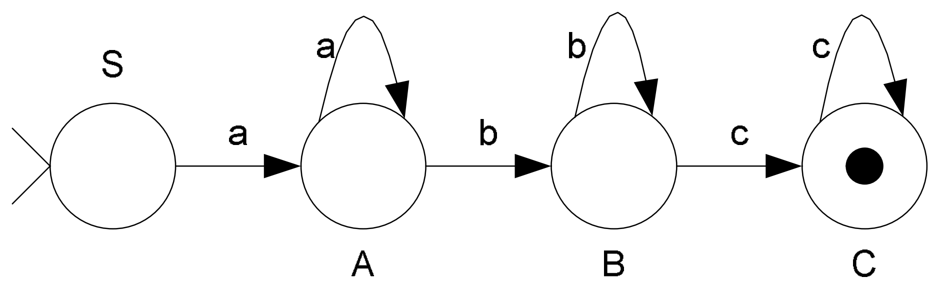
NFA → RG: Example 2

- Step d1: introduce rules for final states
  - V = {S, A, B, C}
  - Σ = {a, b, c}
  - S = S
  - R = {(S, aA), (A, aA), (A, bB), (B, bB), (B, cC), (C, cC), **(C, *e*)**}



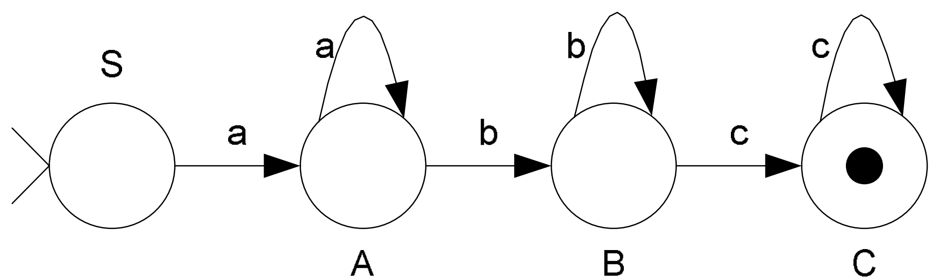
NFA → RG: Example 2

- Operation for abc:
  - (S, abc) |- (A, bc) |- (B, c) |- (C, *e*)
  - S => aA => abB => abcC => abc



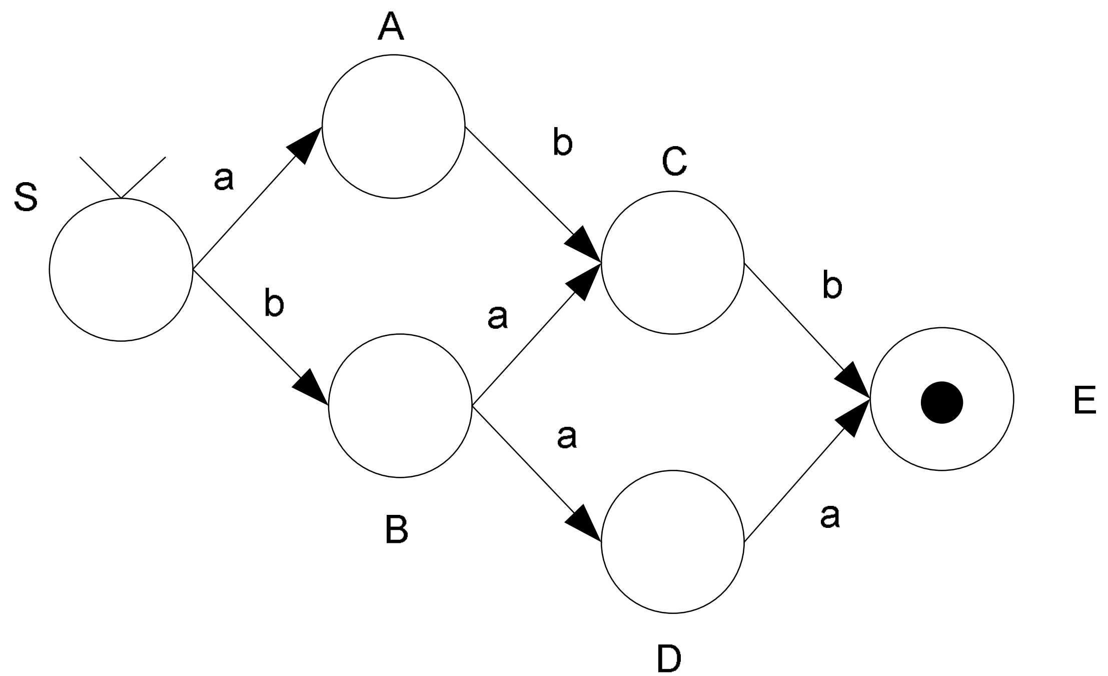
NFA → RG: Example 2

- Operation for aabbcc:
  - (S, aabbcc) |- (A, abbcc) |- (A, bbcc) |- (B, bcc) |- (B, cc) |- (C, c) |- (C, *e*)
  - S => aA => aaA => aabB => aabbB => aabbcC => aabbccC => aabbcc



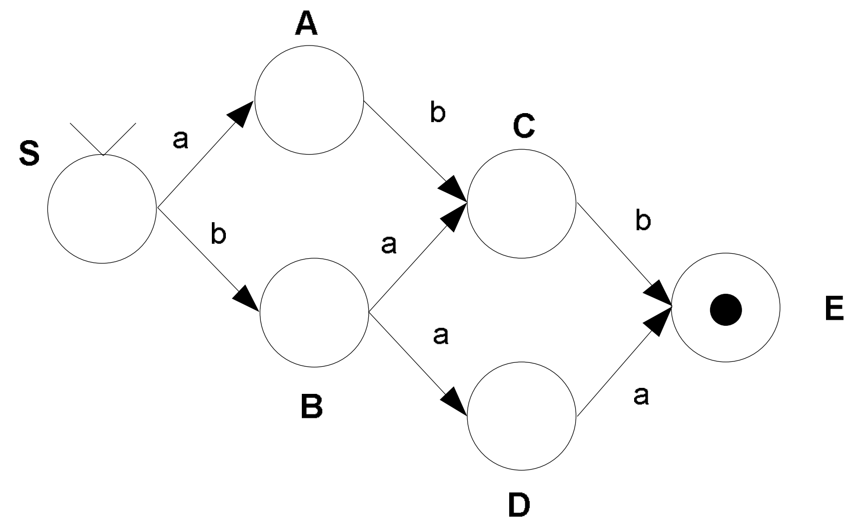
NFA → RG: Example 3

- Construct RG G such that L(G) = L(M)!



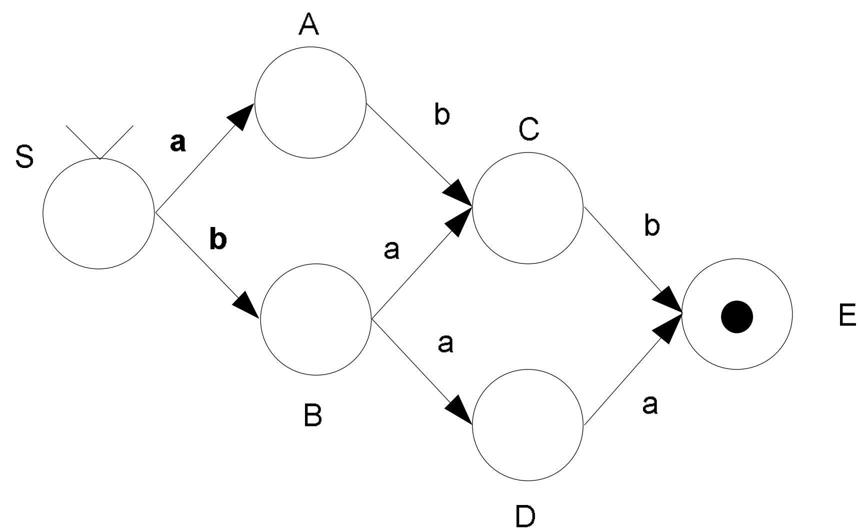
NFA → RG: Example 3

- Steps a: introduce non-terminals
  - V = {**S**, **A**, **B**, **C**, **D**, **E**}



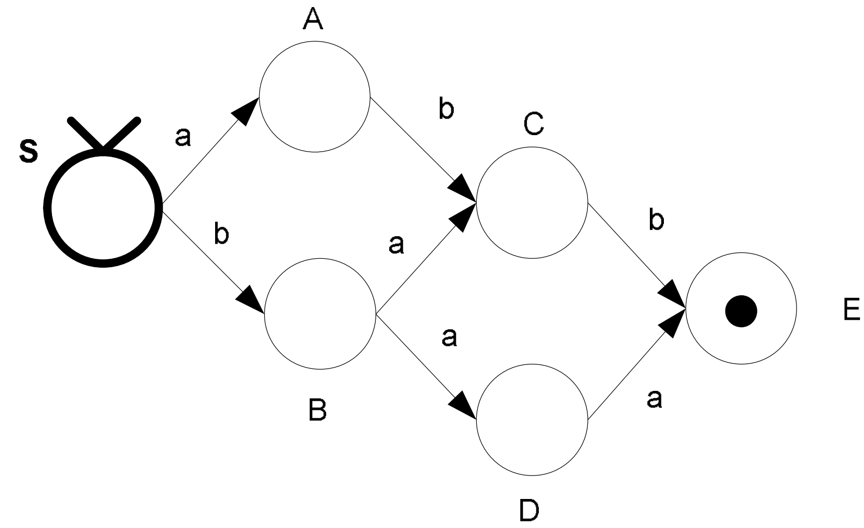
NFA → RG: Example 3

- Step: introduce Σ
  - V = {S, A, B, C, D, E}
  - **Σ = {a, b}**



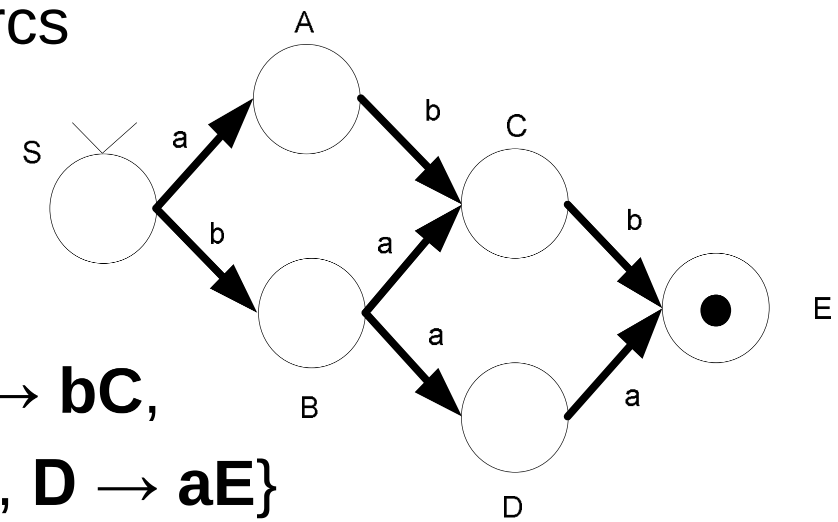
NFA → RG: Example 3

- Step: introduce starting non-terminal
  - V = {S, A, B, C, D, E}
  - Σ = {a, b}
  - S = **S**



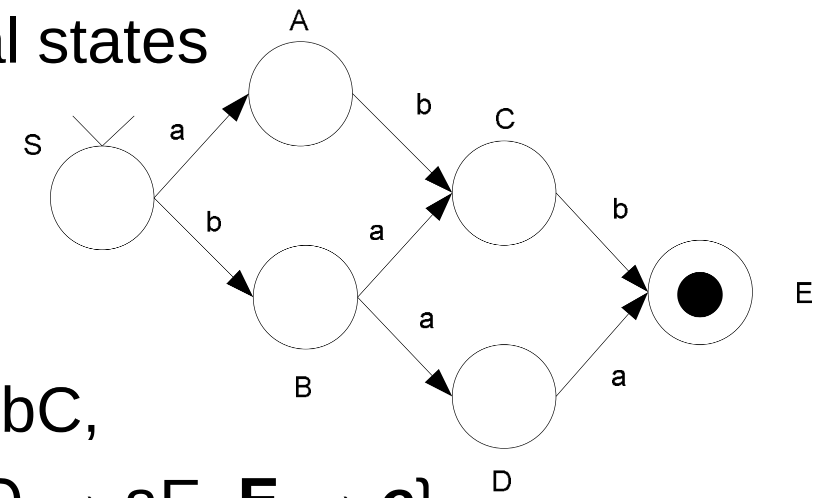
NFA → RG: Example 3

- Steps b: introduce rules for arcs
  - V = {S, A, B, C, D, E}
  - Σ = {a, b}
  - S = S
  - R = {**S → aA**, **S → bB**, **A → bC,** **B → aC**, **B → aD**, **C → bE**, **D → aE**}



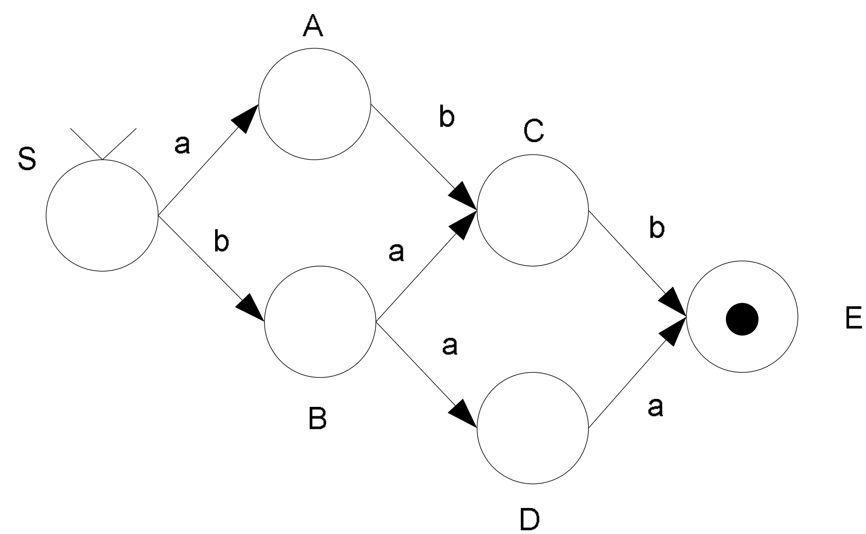
NFA → RG: Example 3

- Steps c: introduce rules for final states
  - V = {A, B, C, D, E}
  - Σ = {a, b}
  - S = S
  - R = {S → aA, S → bB, A → bC,
    B → aC, B → aD, C → bE, D → aE, **E** → ***e***}



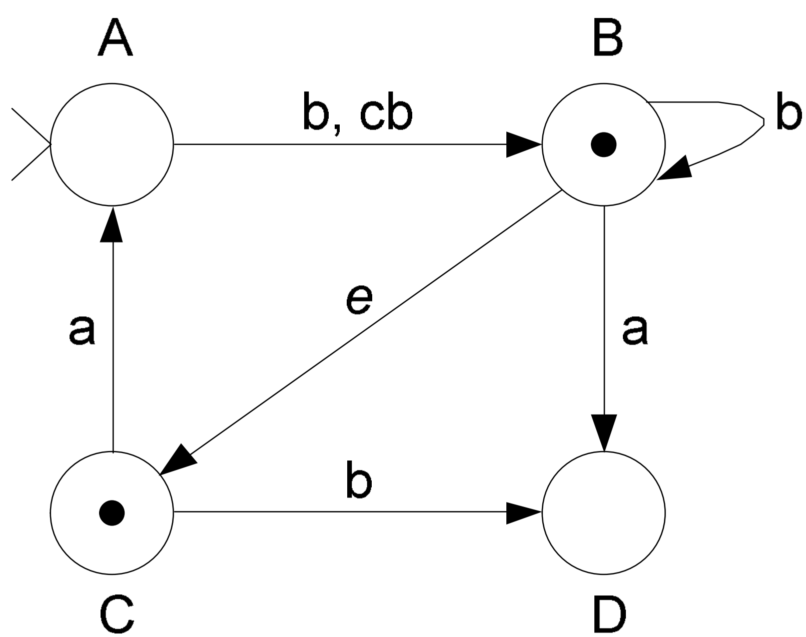
NFA → RG: Example 3

- $R = \{S \rightarrow aA, S \rightarrow bB, A \rightarrow bC, B \rightarrow aC, B \rightarrow aD, C \rightarrow bE, D \rightarrow aE, E \rightarrow e\}$
- Operation for abb:
  - (S, abb) |- (A, bb) |- (C, b) |-(E, *e*)
  - S => aA => abC => abbE => abb



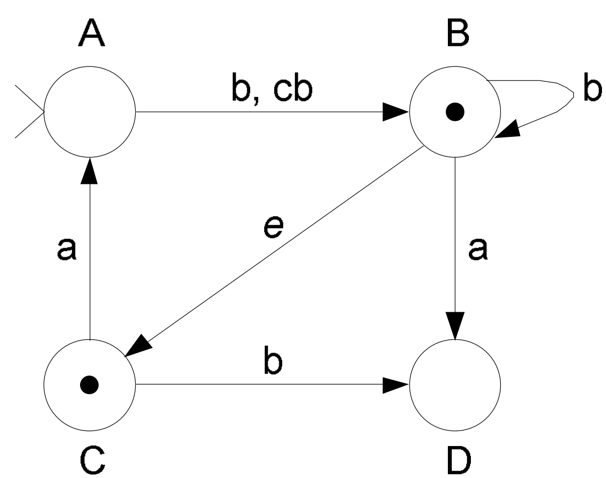
NFA → RG: Example 4

- Construct RG G such that L(G) = L(M)!



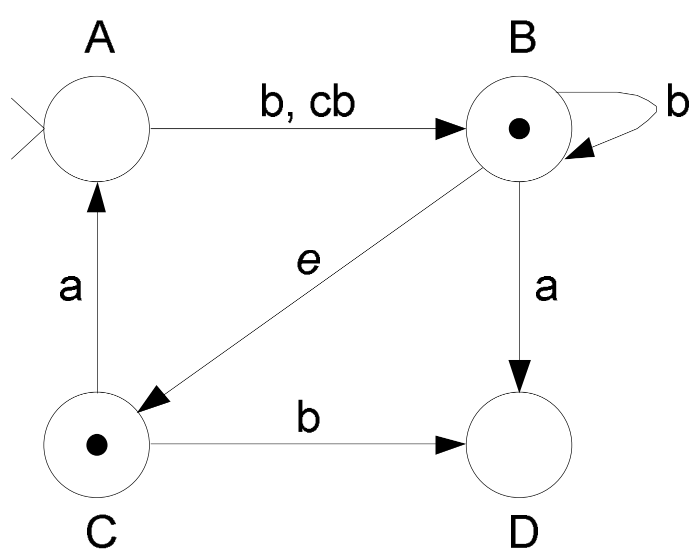
NFA → RG: Example 4

- Solution:
  - V = {A, B, C, D}
  - Σ = {a, b, c}
  - S = A
  - R = {A → bB | cbB, B → bB | C | aD | *e*, C → aA | bD | *e*}



NFA → RG: Example 4

- R = {A → bB | cbB, B → bB | C | aD | *e*, C → aA | bD | *e*}
- Operation for cbb:
  - (A, cbb) |- (B, b) |- (B, *e*)
  - A => cbB => cbbB => cbb



$L \rightarrow RG$: Example 1

- Give RG G such that $L(G) = \{a^n \in \Sigma^*, n \geq 0\}$!
- $G = (\{a, N\}, \{a\}, R, S)$
  - $R = \{S \rightarrow aS \mid e\}$
  - derivation: S => aS => aaS => aa

$L \rightarrow RG$: Example 2

- Give RG G such that $L(G) = \{ba^n \in \Sigma^*, n \geq 0\}$!
- $G = (\{N, a, b\}, \{a, b\}, R, S)$
  - $R = \{S \rightarrow bN, N \rightarrow aN \mid e\}$
  - derivation: S => bN => baN => baaN => baa

L $\rightarrow$ RG: Example 3

- Give RG G such that L(G) = $\{cba^n b^2 \in \Sigma^*, n \geq 0\}$!
- G = ({N, a, b, c}, {a, b, c}, R, S)
  - L(G) = $\{cb(a^n)b^2 \in \Sigma^*, n \geq 0\}$
  - R = {S $\rightarrow$ cbNbb, N $\rightarrow$ aN | *e*}
  - derivation: S => cbNbb => cbaNbb => cbaaNbb => cbaabb

$L \rightarrow RG$: Example 4

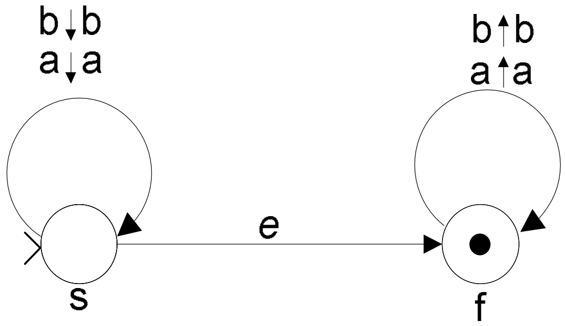
- Give RG G such that $L(G) = \{ac^{m+3}a^{n+1}c^2 \in \Sigma^*, n, m \geq 0\}$!
- $G = (\{N, M, a, b, c\}, \{a, b, c\}, R, S)$
  - $L(G) = \{accc(c^m)a(a^n)c^2 \in \Sigma^*, n, m \geq 0\}$
  - $R = \{S \rightarrow aMcccNacc, M \rightarrow cM \mid e, N \rightarrow aN \mid e\}$
  - derivation: S => aMcccNacc => acMcccNacc => accccNacc => accccaNacc => acccccaacc

Element of the Theory of the Computation

Lecture 8

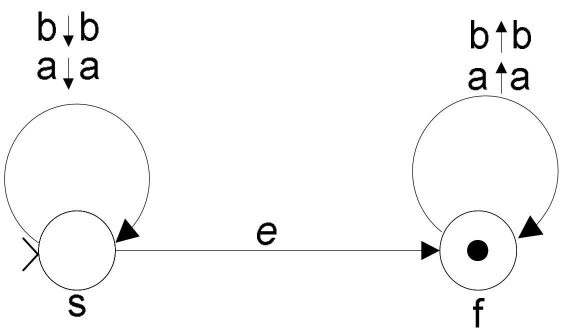
PDA computation

- Give the sate diagram for M!
  - L(M) = $\{ww^R : w \in \{a, b\}^*\}$
  - K = {s, f}
  - Σ = {a, b}
  - F = {f}
  - Δ:
    - ((s, a, *e*), (s, a))
    - ((s, b, *e*), (s, b))
    - ((s, *e*, *e*), (f, *e*))
    - ((f, a, a), (f, *e*))
    - ((f, b, b), (f, *e*))



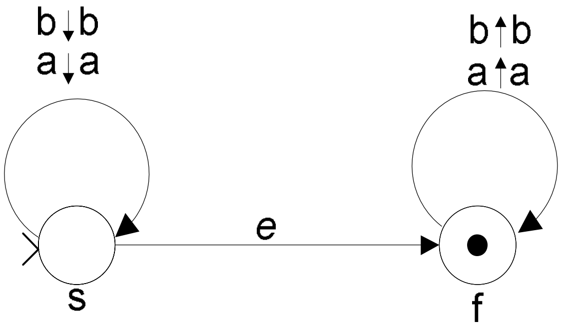
PDA computation

- Give the computation of M for string "abaaba"!



- Result:
  - (s, abaaba, *e*) |- (s, baaba, a) |- (s, aaba, ba) |- (s, aba, aba) |- (f, aba, aba) |- (f, ba, ba) |- (f, a, a) |- (f, *e*, *e*)

PDA computation



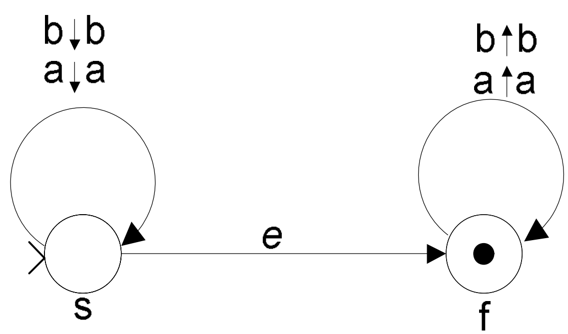
- Give the computation of M for string "bbbb" using the same M as before!

State	Unread input	Stack
s	bbbb	*e*
s	bbb	b
s	bb	bb
f	bb	bb
f	b	b
f	*e*	*e*

PDA computation

State	Unread input	Stack
s	bbbb	*e*
s	bbb	b
s	b	bbb
f	b	bbb
f	*e*	bb
f	*e*	*e*

- Does the above table represent a computation of M?
  - yes
- Is bbbb accepted or rejected?
  - accepted



PDA: Example 1

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$

PDA: Example 1

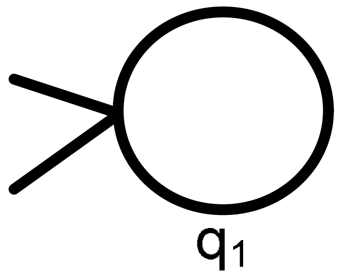
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 'a' excess, 'b' lack

'-' — 'b' excess, 'a' lack



PDA: Example 1

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack

PDA: Example 1

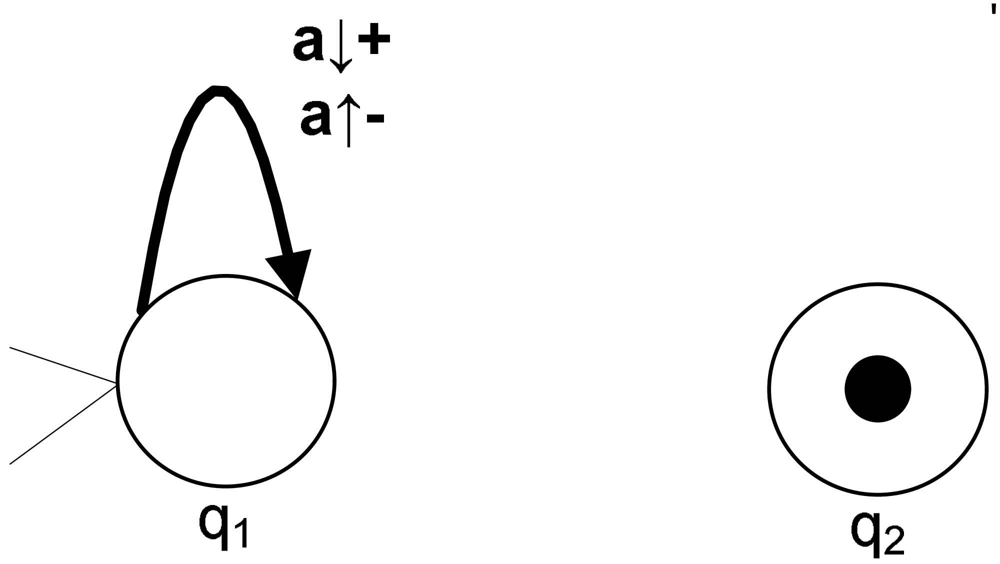
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack



PDA: Example 1

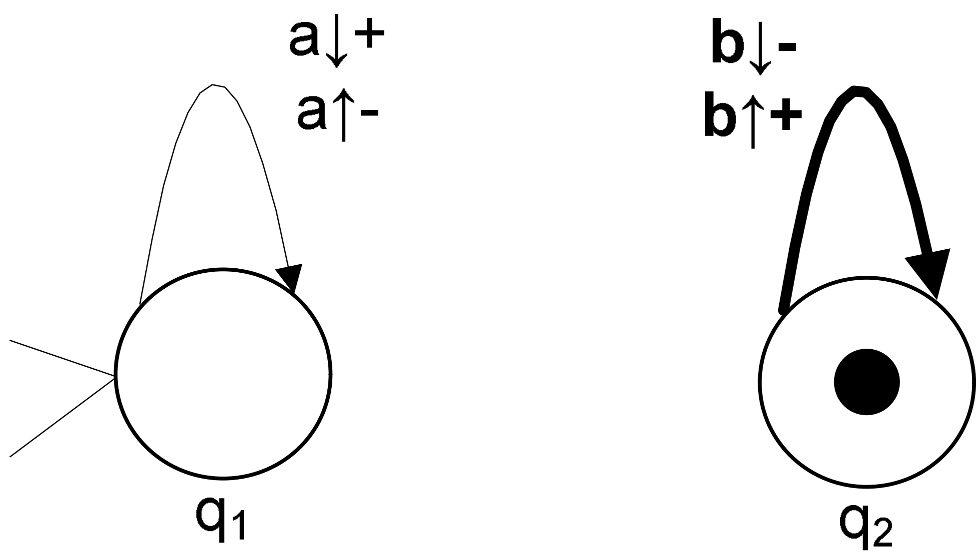
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack



PDA: Example 1

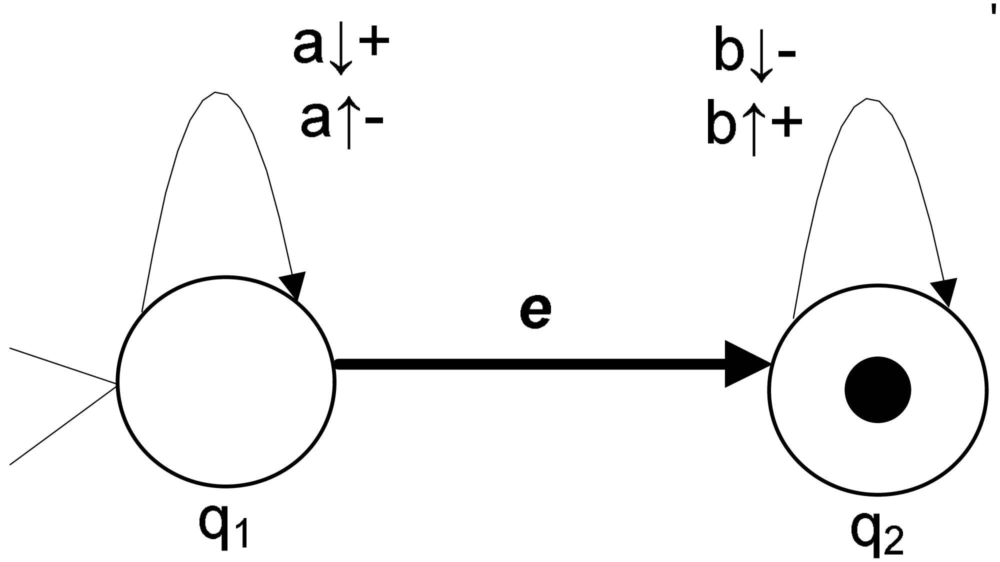
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack



PDA: Example 1

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$

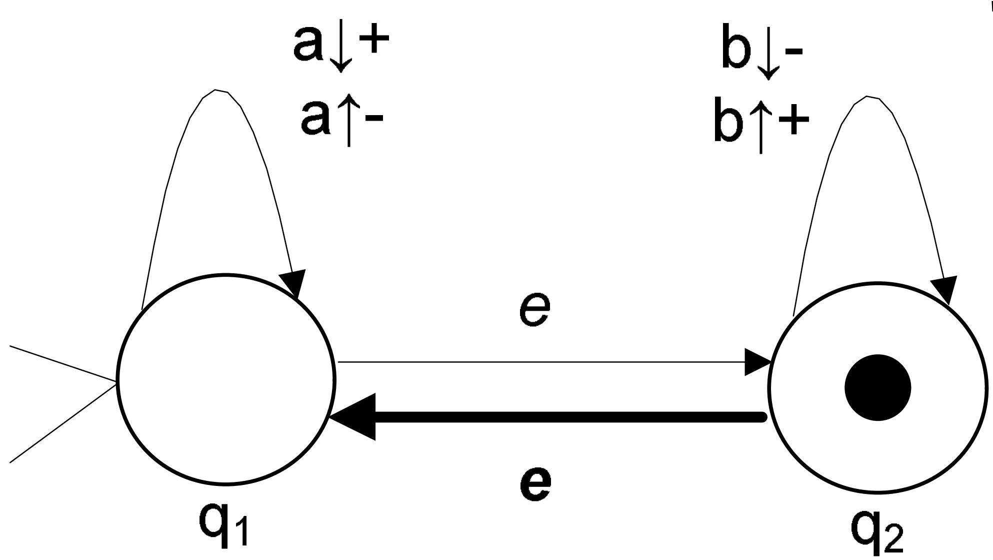
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack

- Solution:



PDA: Example 1

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = |b|\}$

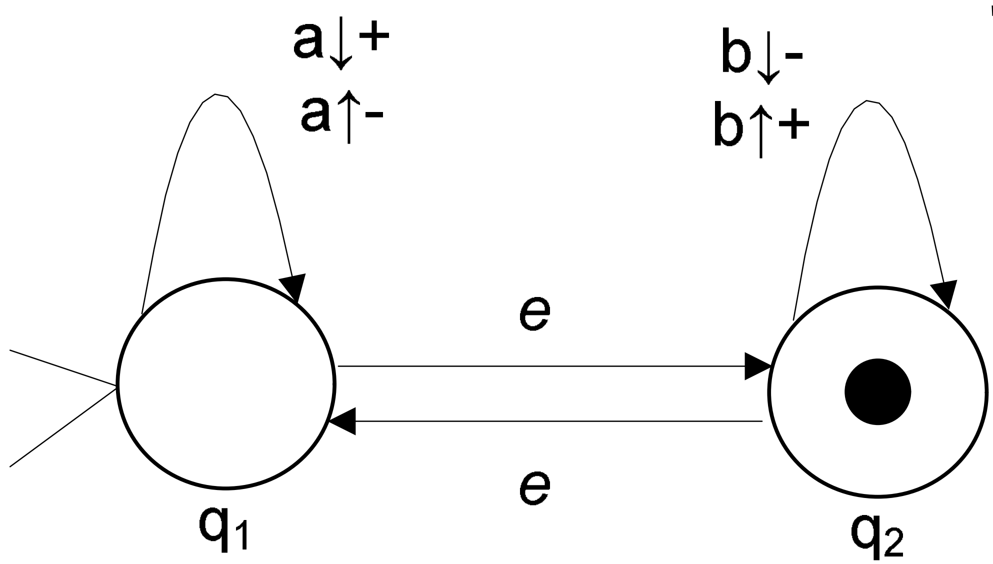
q_1 — 'a' reader state

q_2 — 'b' reader state

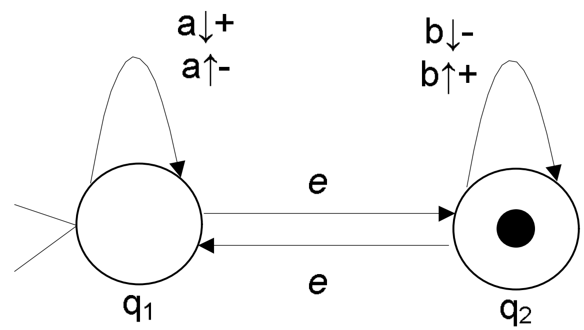
'+' — 1 'a' excess, 1 'b' lack

'-' — 1 'b' excess, 1 'a' lack

- Solution:



PDA: Example 1



State	Unread input	Stack
q_1	abbaba	*e*
q_1	bbaba	+
q_2	bbaba	+
q_2	baba	*e*
q_2	aba	-
q_1	aba	-
q_1	ba	*e*
q_2	ba	*e*
q_2	a	-
q_1	a	-
q_1	*e*	*e*
q_2	*e*	*e*

PDA: Example 2

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$

PDA: Example 2

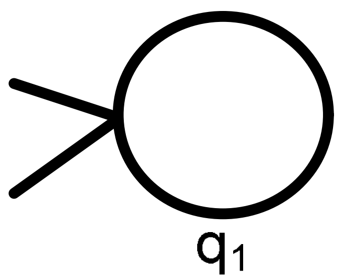
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$
- Remark:
  - '+' in the stack means 1/3 'b' lack
  - 1/3 'b' cannot be read, but you can read two additional 'a'; then "+++" means 3/3 'b' lack
- Solution:



PDA: Example 2

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack

PDA: Example 2

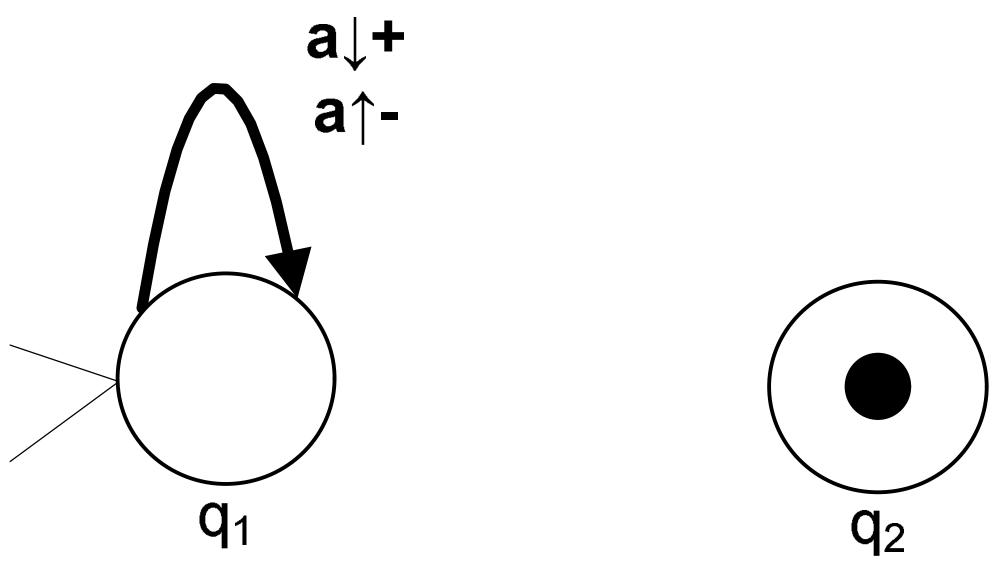
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack



PDA: Example 2

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | |a| = 3|b|}

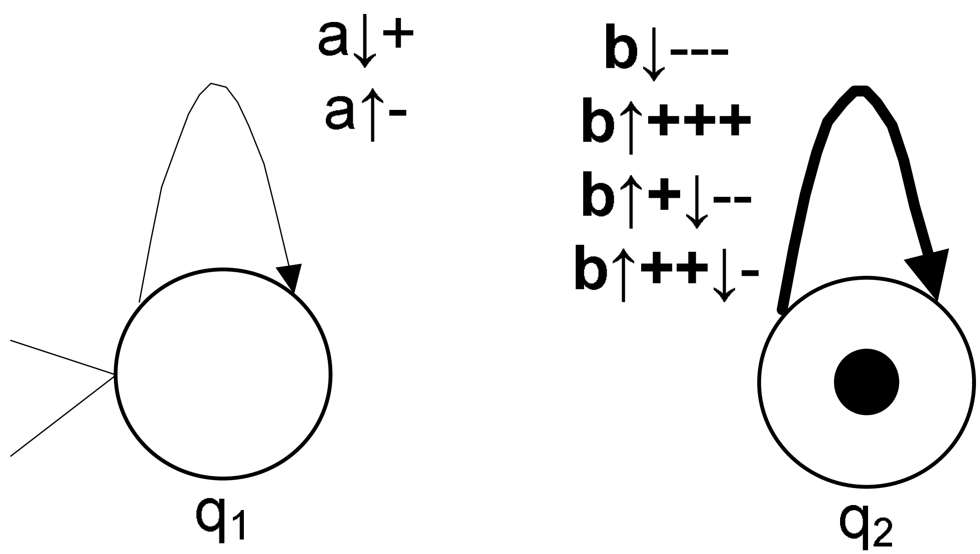
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack

- Solution:



push 3*1/3 'b' excess

pop 3*1/3 'b' lack

pop 1*1/3 'b' lack and push 2*1/3 'b' excess

pop 2*1/3 'b' lack and push 1*1/3 'b' excess

PDA: Example 2

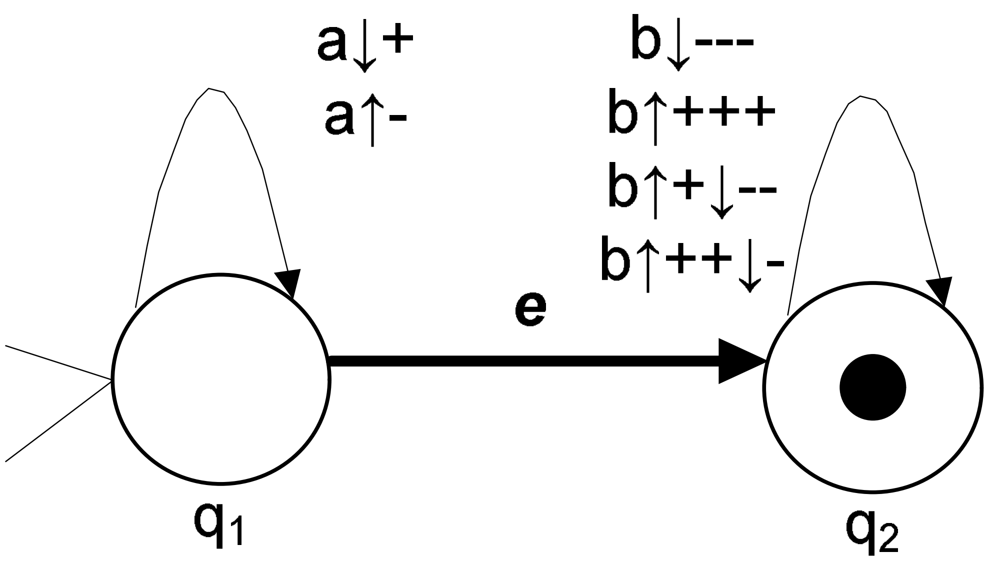
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack



PDA: Example 2

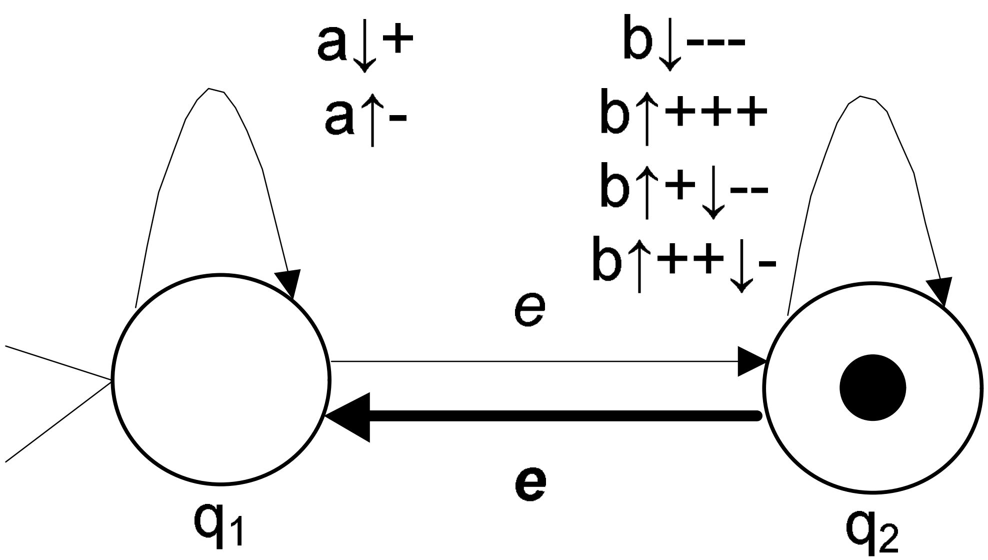
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid |a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack



PDA: Example 2

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | |a| = 3|b|}

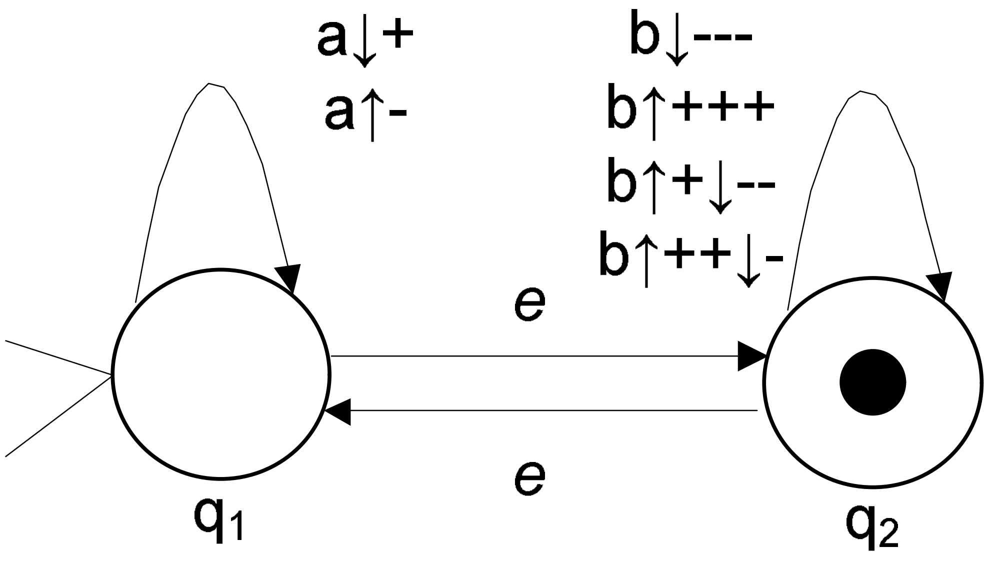
q_1 — 'a' reader state

q_2 — 'b' reader state

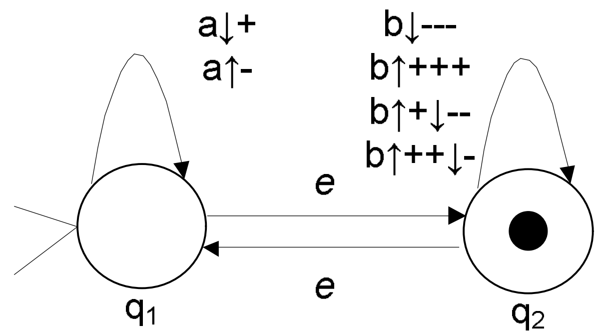
'+' — 1 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1 'a' lack

- Solution:



PDA: Example 2



State	Unread input	Stack
q_1	aaba	*e*
q_1	aba	+
q_1	ba	++
q_2	ba	++
q_2	a	-
q_1	a	-
q_1	*e*	*e*
q_2	*e*	*e*

PDA: Example 3

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | 2|a| = 3|b|}
- Let '+' = 1/2 'a' excess
  - if '+' is in the stack then the 2|a| = 3|b| equation does not hold
  - instead 2(|a| - 1/2) = 3|b| is true
  - 2|a| - 1 = 3|b|
  - 2|a| = 3|b| + 1
  - 2|a| = 3(|b| + 1/3)
  - '+' also means 1/3 'b' lack
- Let '-' = 1/3 'b' excess
  - '-' also means 1/2 'a' lack

PDA: Example 3

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 2|a| = 3|b|}

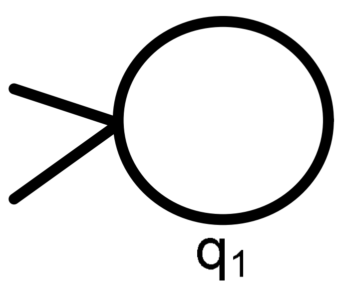
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack

- Solution:



PDA: Example 3

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 2|a| = 3|b|\}$

- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack

PDA: Example 3

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 2|a| = 3|b|}

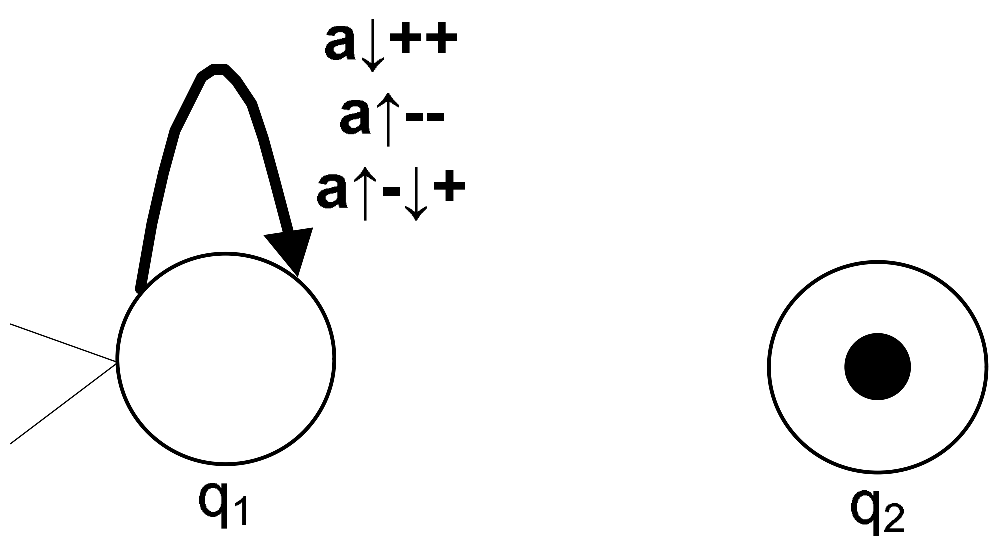
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack

- Solution:



PDA: Example 3

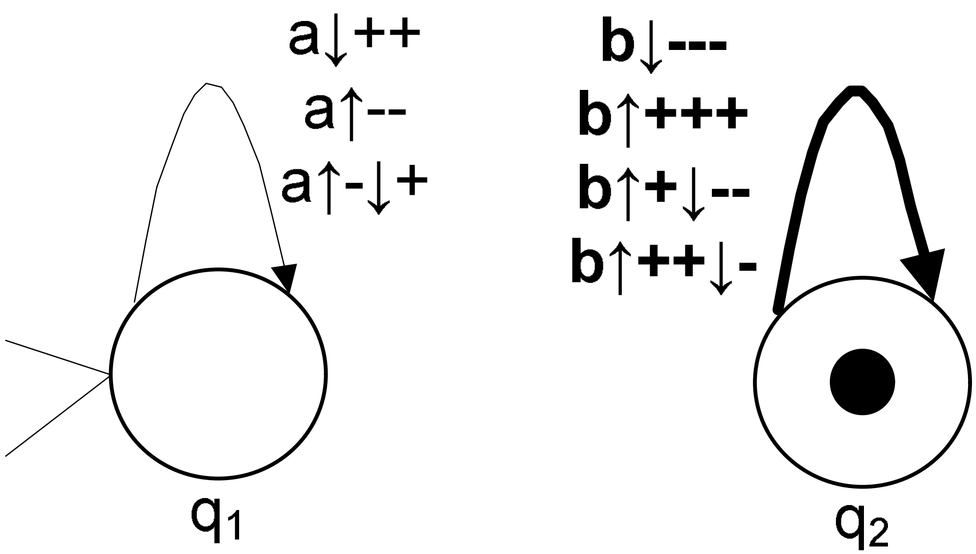
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 2|a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack



PDA: Example 3

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 2|a| = 3|b|}

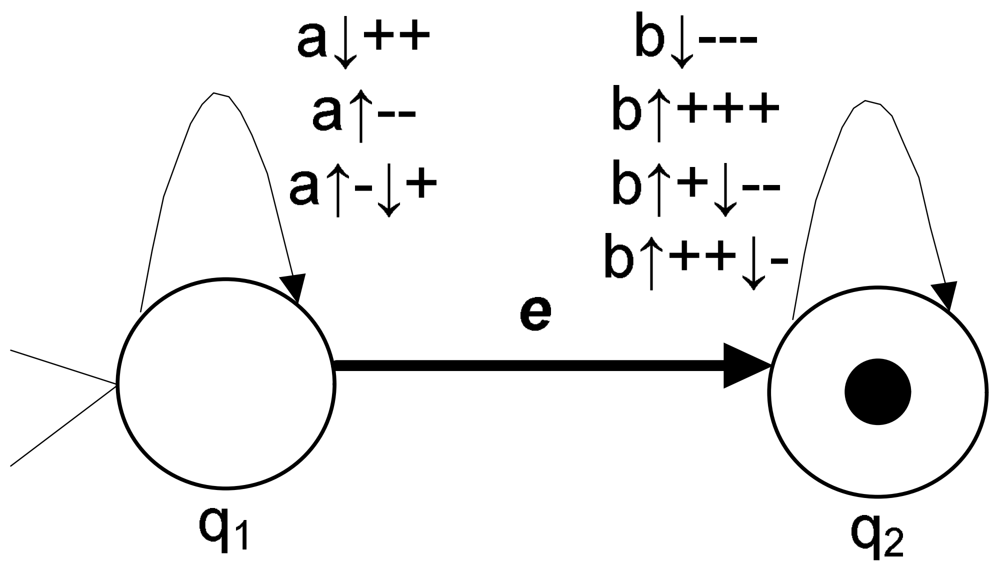
q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack

- Solution:



PDA: Example 3

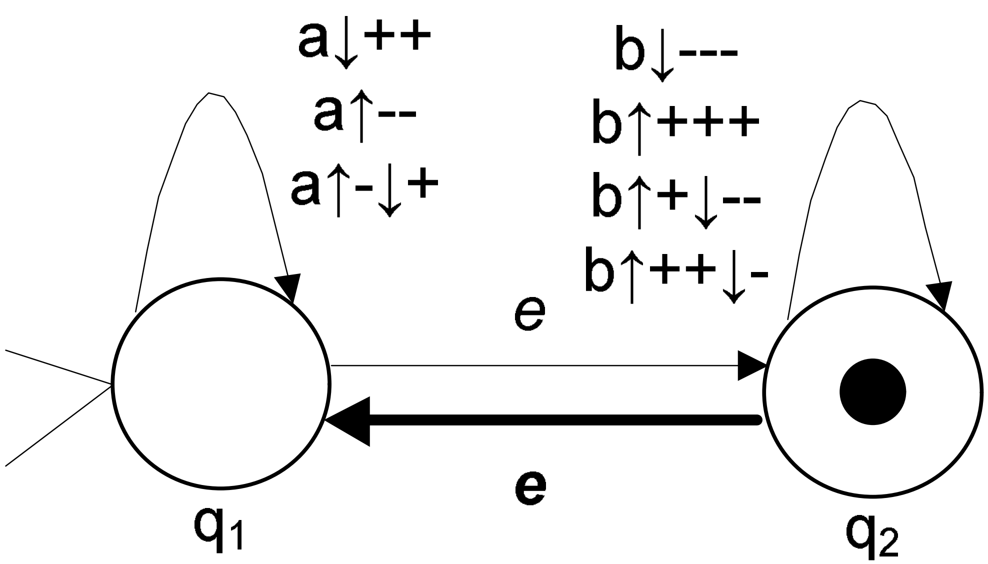
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 2|a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack



PDA: Example 3

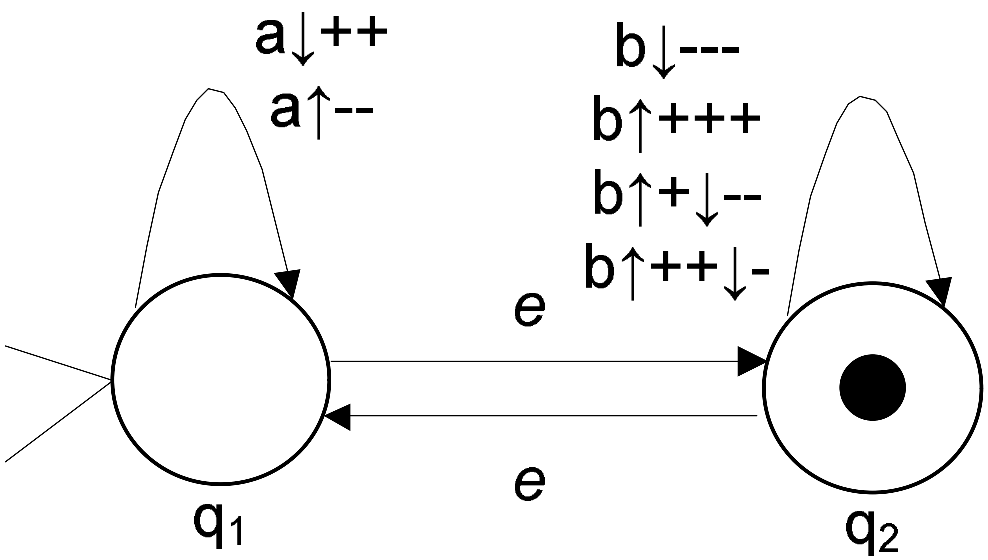
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 2|a| = 3|b|\}$
- Solution:

q_1 — 'a' reader state

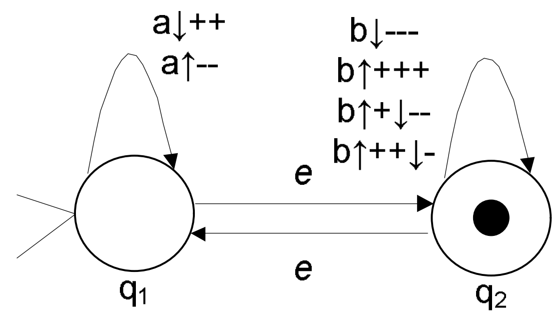
q_2 — 'b' reader state

'+' — 1/2 'a' excess, 1/3 'b' lack

'-' — 1/3 'b' excess, 1/2 'a' lack



PDA: Example 3



State	Unread input	Stack
q_1	aabab	e
q_1	abab	++
q_1	bab	++++
q_2	bab	++++
q_2	ab	+
q_1	ab	+
q_1	b	+++
q_2	b	+++
q_2	e	e

PDA: Example 4

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 3|a| - 2|b| = 4\}$

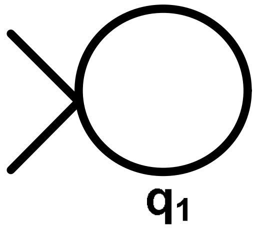
PDA: Example 4

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 3|a| - 2|b| = 4\}$
  - $L = \{w \in \{a, b\}^* \mid 3|a| = 2|b| + 4\}$
- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack

'+' and '-' is interpreted without the +4



PDA: Example 4

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 3|a| - 2|b| = 4}

- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack

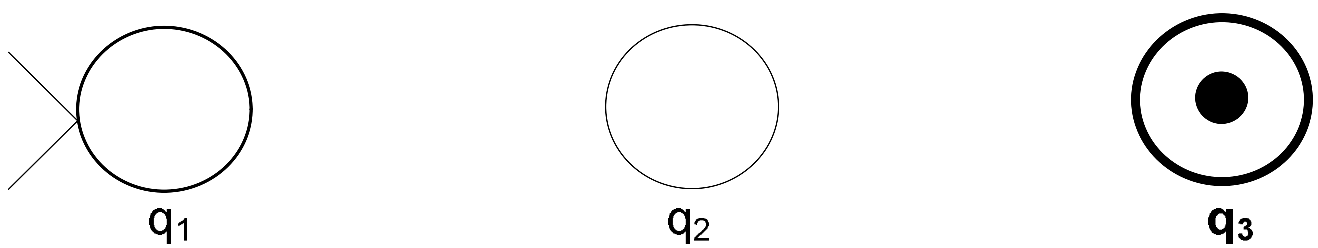
PDA: Example 4

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 3|a| - 2|b| = 4\}$

- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack

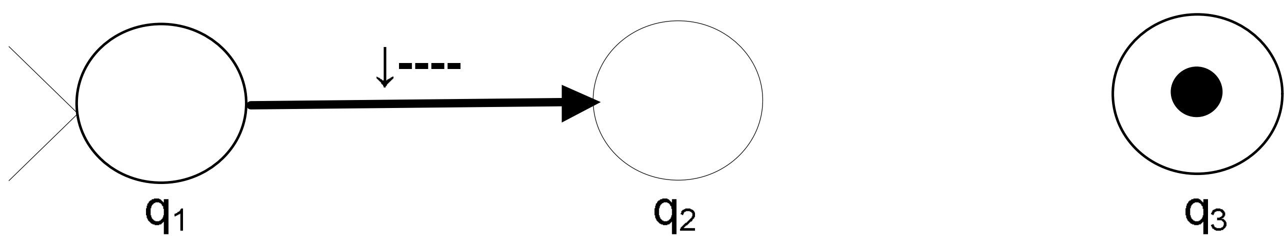


PDA: Example 4

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 3|a| - 2|b| = 4\}$
- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack



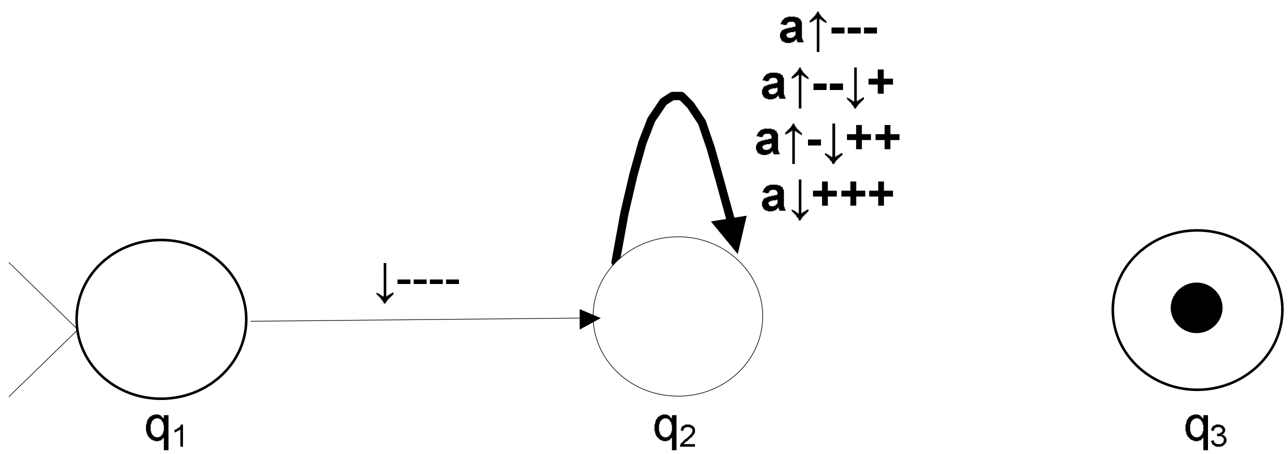
PDA: Example 4

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid 3|a| - 2|b| = 4\}$

- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack



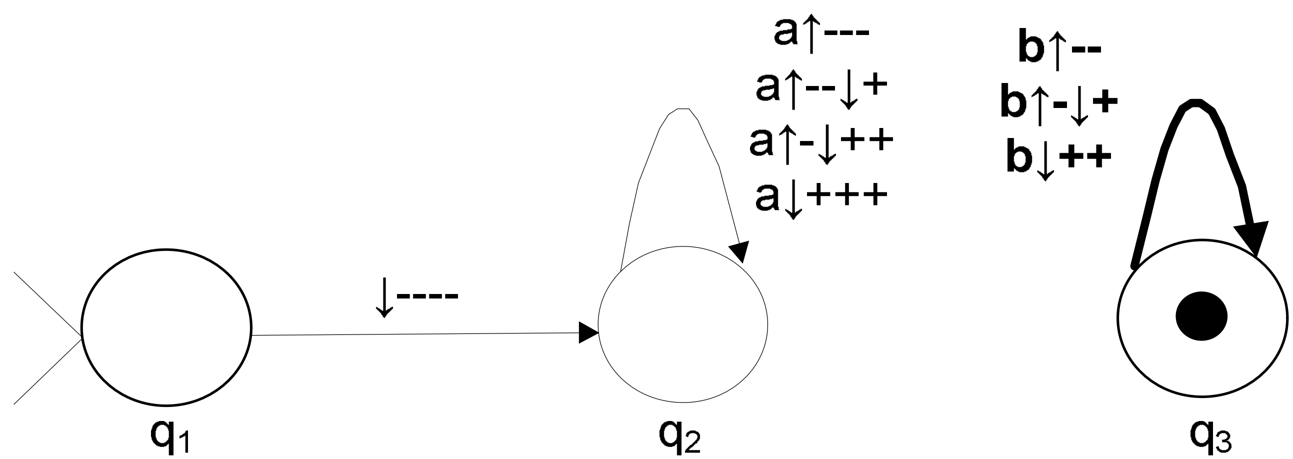
PDA: Example 4

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 3|a| - 2|b| = 4}

- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack

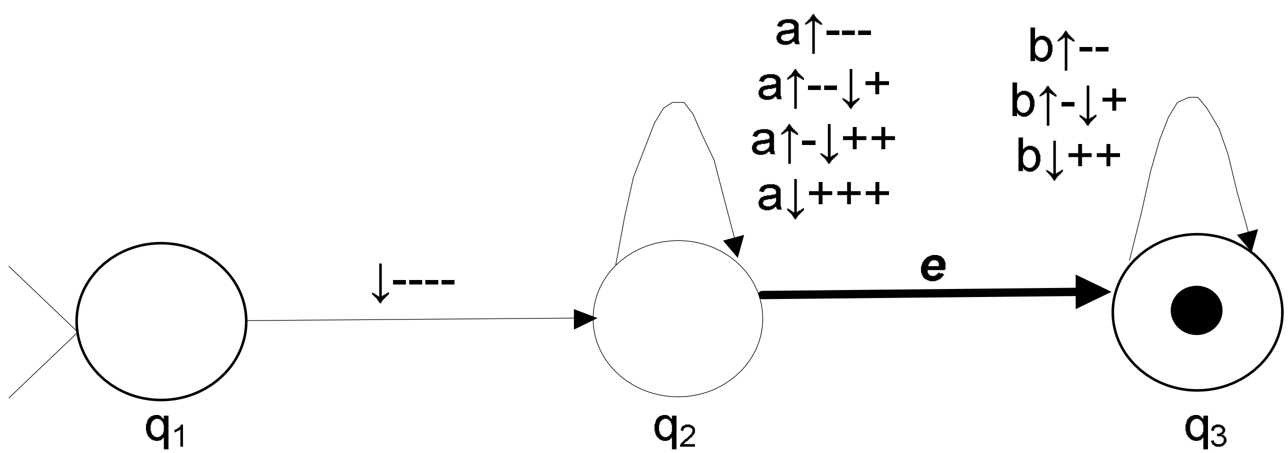


PDA: Example 4

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 3|a| - 2|b| = 4}
- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack



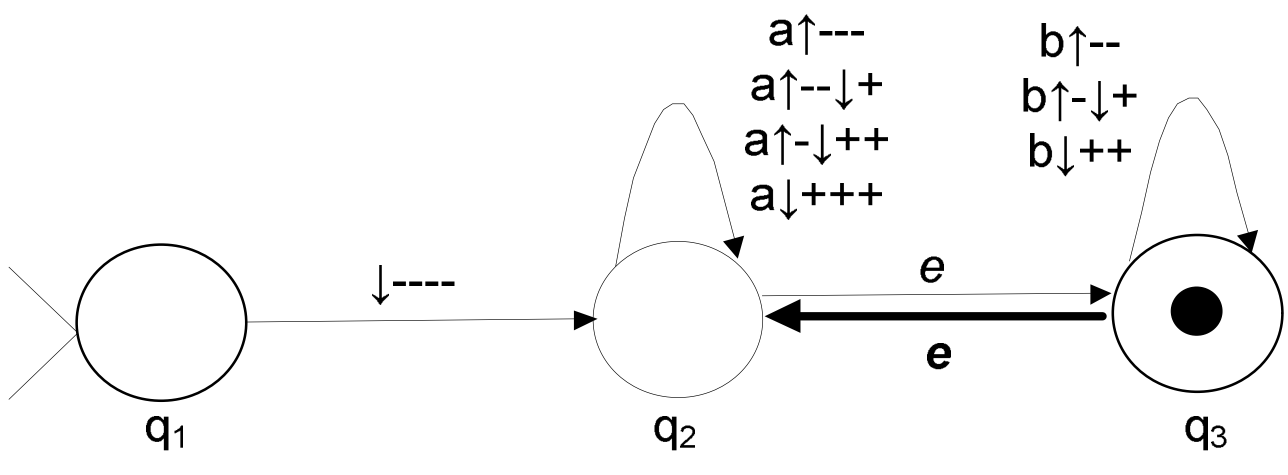
PDA: Example 4

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 3|a| - 2|b| = 4}

- Solution:

'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack



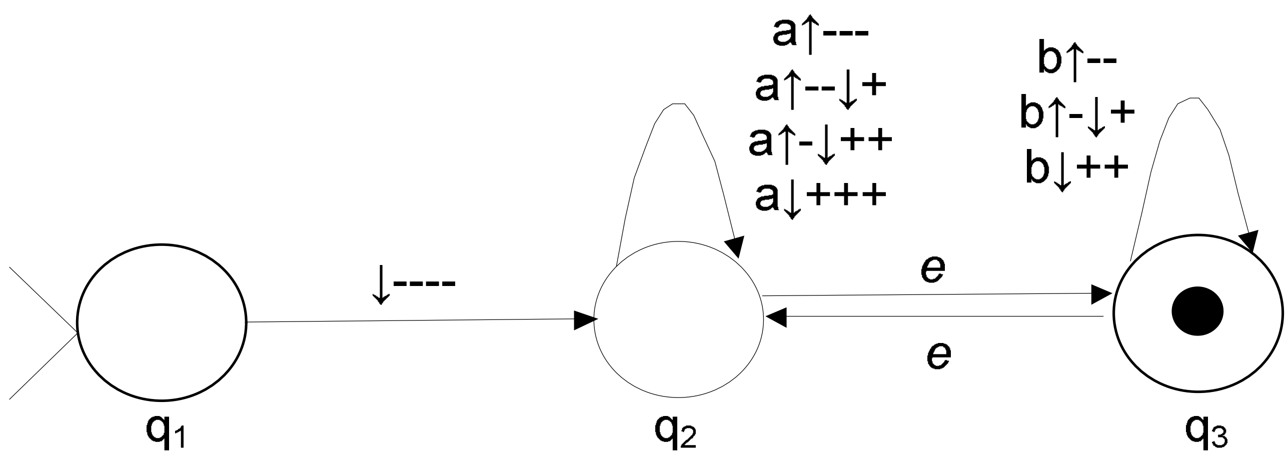
PDA: Example 4

- Give PDA M such that L(M) = L!
  - L = {w ∈ {a, b}* | 3|a| - 2|b| = 4}

- Solution:

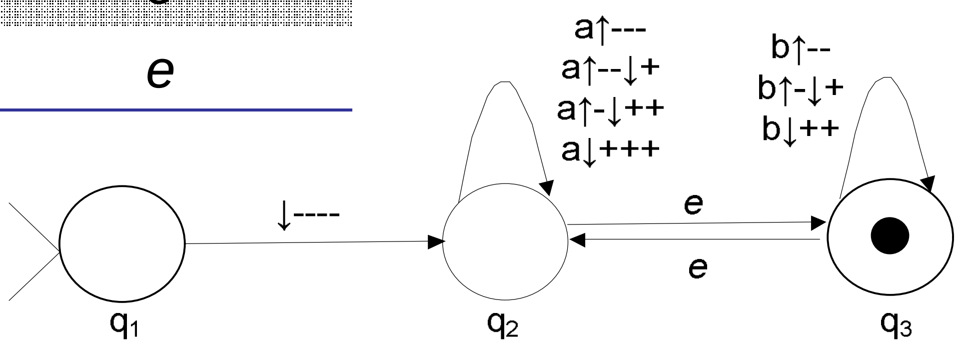
'+' — 1/3 'a' excess, 1/2 'b' lack

'-' — 1/2 'b' excess, 1/3 'a' lack



PDA: Example 4

State	Unread input	Stack
q_1	aba	e
q_2	aba	----
q_2	ba	-
q_3	ba	-
q_3	a	---
q_2	a	---
q_2	e	e
q_3	e	e



PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

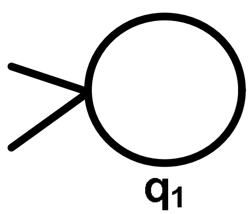
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack and 1 "d" lack

'-' is not used because the order of the characters is fixed here



PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack and 1 "d" lack

PDA: Example 5

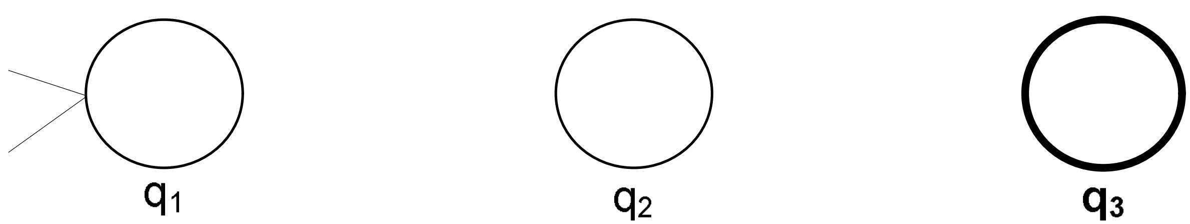
- Give PDA M such that L(M) = L!
  – $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack and 1 "d" lack



PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack

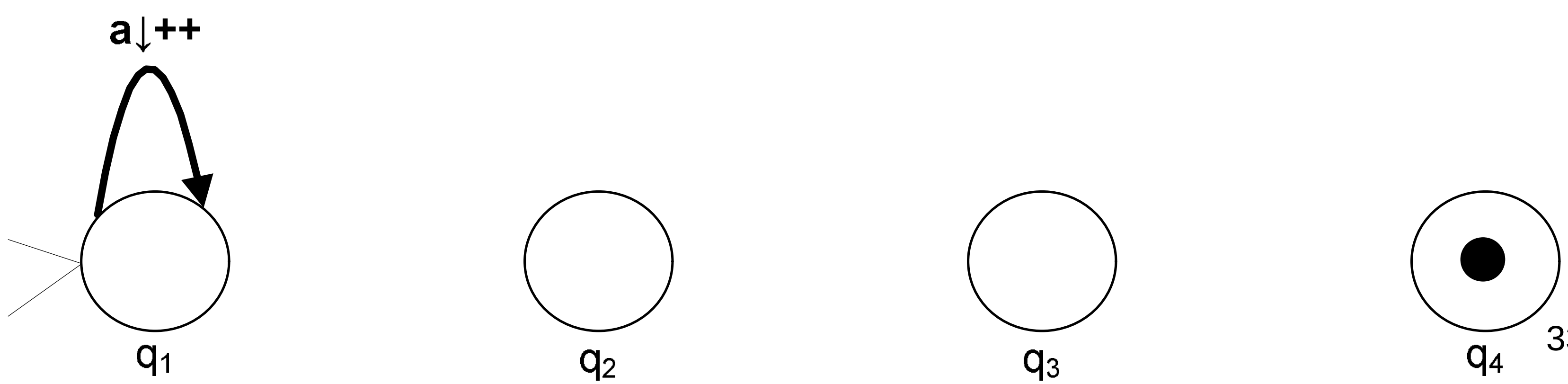
PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



PDA: Example 5

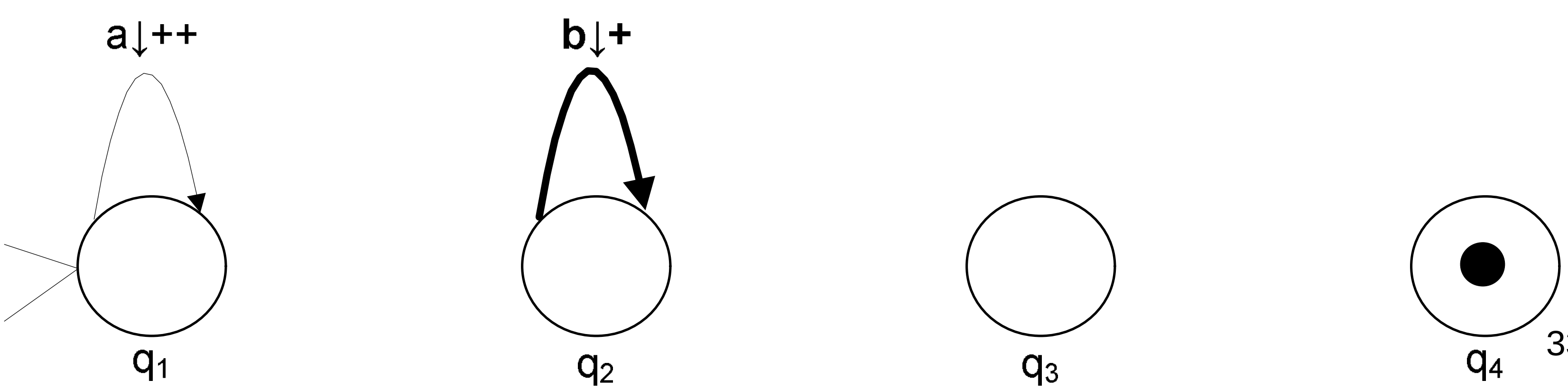
- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



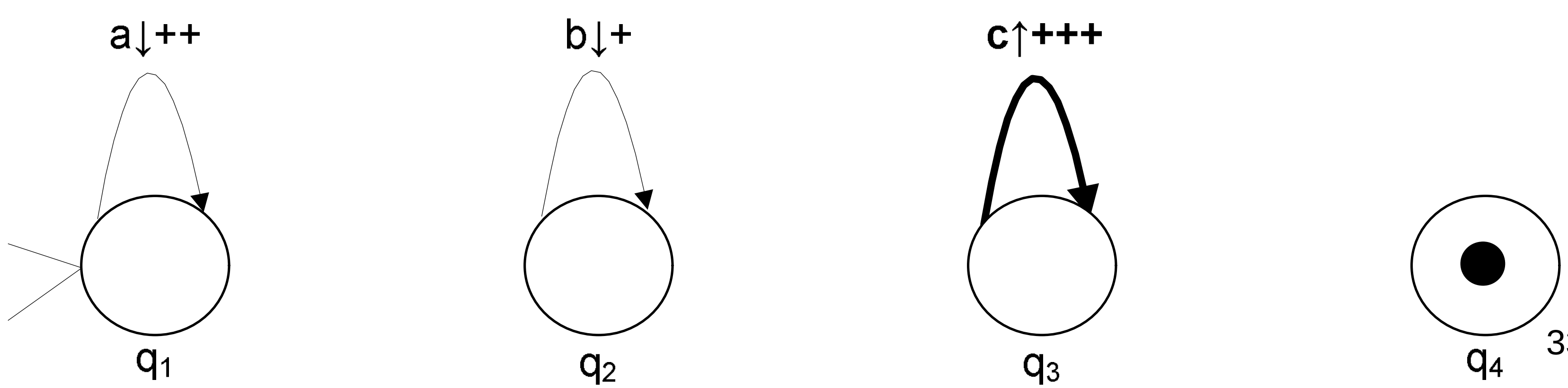
PDA: Example 5

- Give PDA M such that L(M) = L!
  - L = $\{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



PDA: Example 5

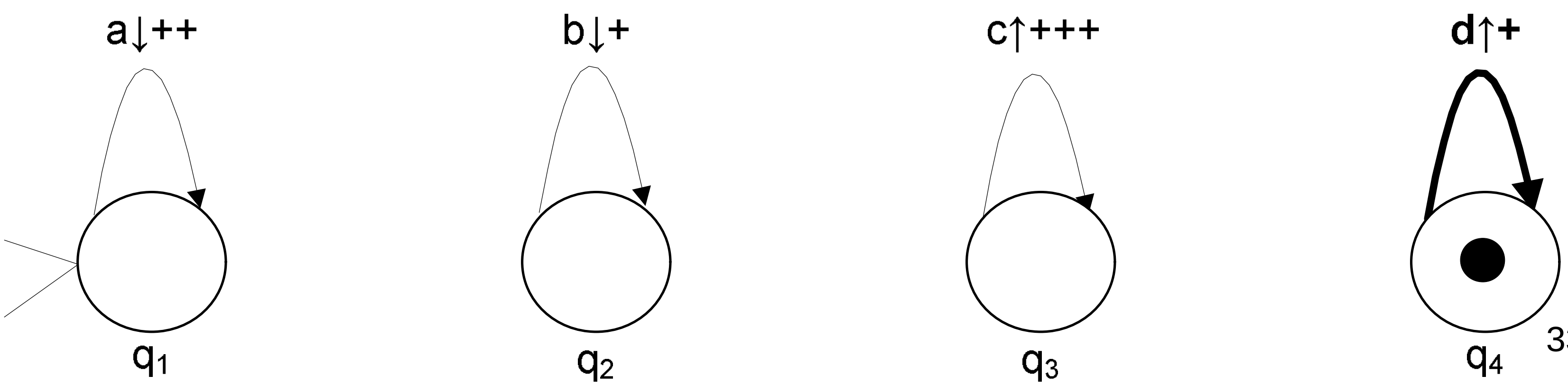
- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



PDA: Example 5

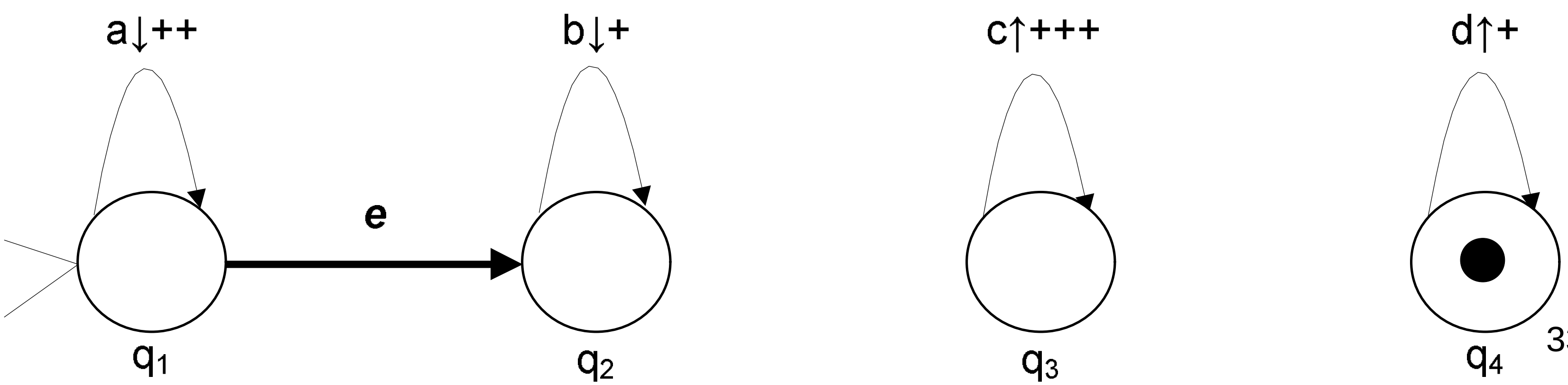
- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



PDA: Example 5

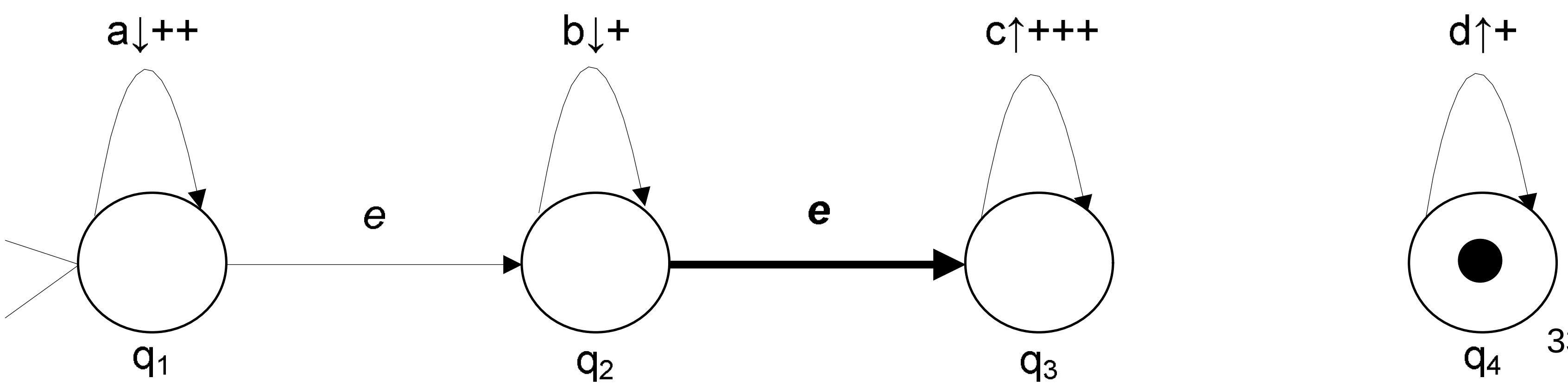
- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



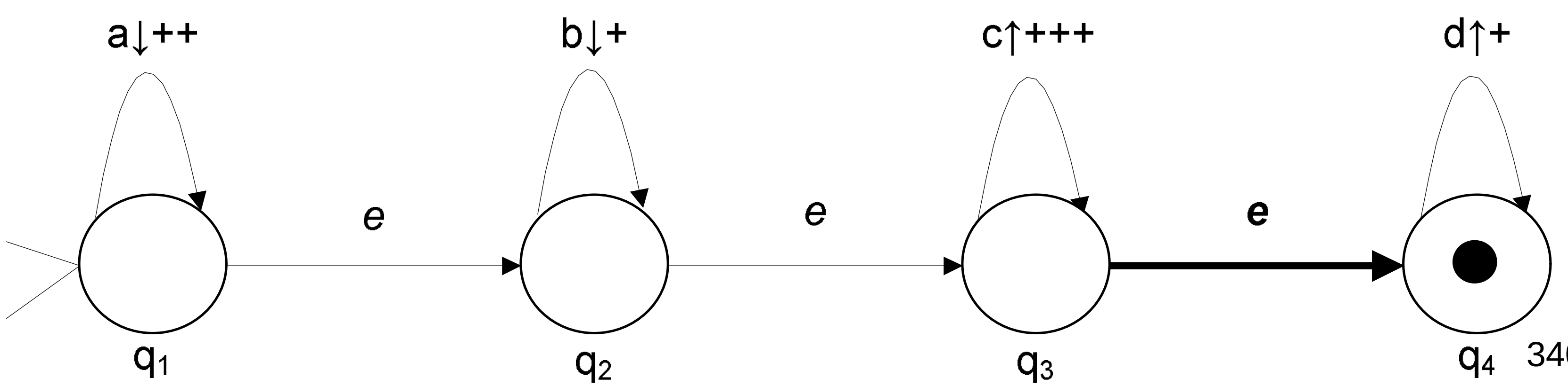
PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



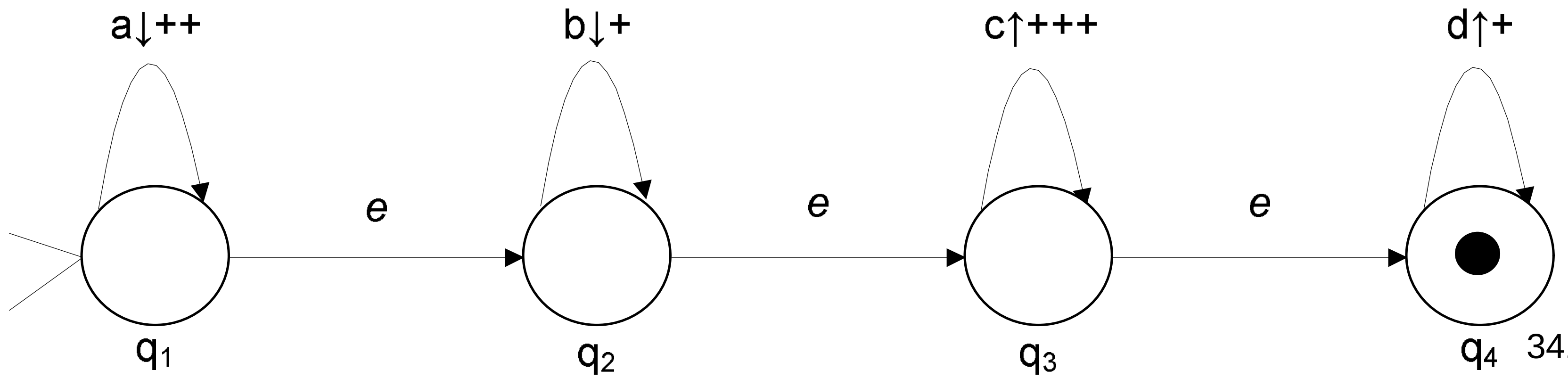
PDA: Example 5

- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + m = 3p + q\}$
- Solution:

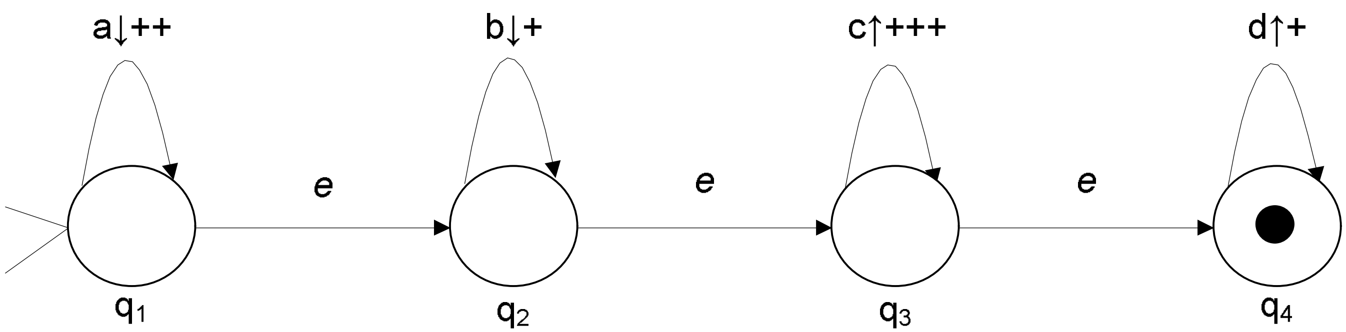
q_1 — 'a' reader state q_2 — 'b' reader state

q_3 — 'c' reader state q_4 — 'd' reader state

'+' — 1/2 "a" excess, 1 "b" excess, 1/3 "c" lack, 1 "d" lack



State	Unread input	Stack
q_1	aaabccd	e
q_1	aabccd	$+^2$
q_1	abccd	$+^4$
q_1	bccd	$+^6$
q_2	bccd	$+^6$
q_2	ccd	$+^7$
q_3	ccd	$+^7$
q_3	cd	$+^4$
q_3	d	$+^1$
q_4	d	$+^1$
q_4	e	e



PDA: Example 6

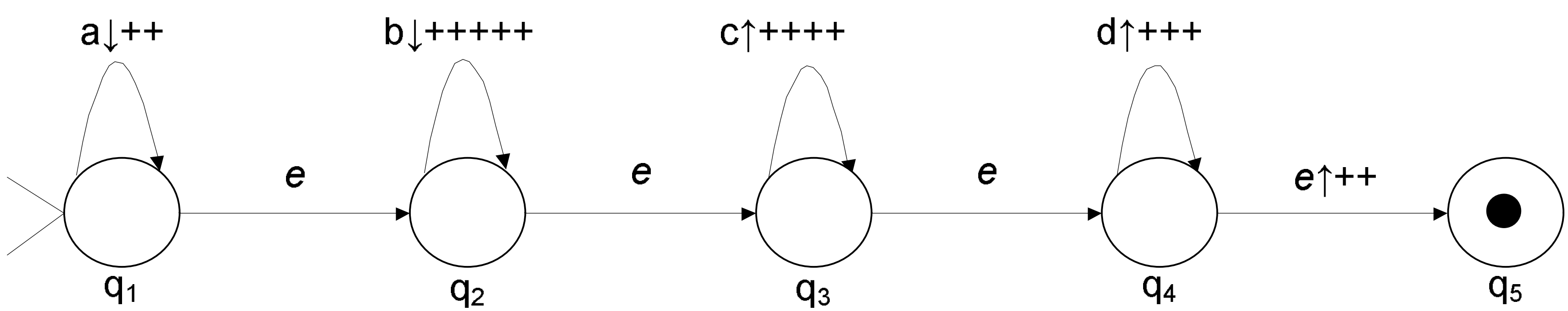
- Give PDA M such that L(M) = L!
  - $L = \{w \in a^n b^m c^p d^q \mid 2n + 5m = 4p + 3q + 2\}$
- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state q_3 — 'c' reader state

q_4 — 'd' reader state q_5 — state for +2

'+' — 1/2 "a" excess, 1/5 "b" excess, 1/4 "c" lack, 1/3 "d" lack

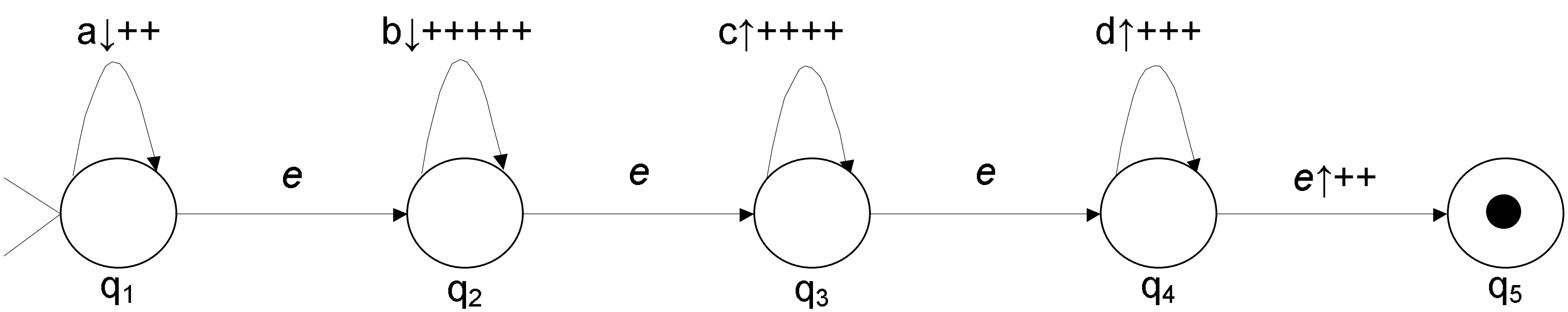
'+' is interpreted without the +2 constant



PDA: Example 6

- Computation for "aaaabccd":

  $(q_1, \text{aaaabccd}, e) \vdash (q_1, \text{aaabccd}, +^2) \vdash$
  $(q_1, \text{aabccd}, +^4) \vdash (q_1, \text{abccd}, +^6) \vdash (q_1, \text{bccd}, +^8) \vdash$
  $(q_2, \text{bccd}, +^8) \vdash (q_2, \text{ccd}, +^{13}) \vdash (q_3, \text{ccd}, +^{13}) \vdash$
  $(q_3, \text{cd}, +^9) \vdash (q_3, \text{d}, +^5) \vdash (q_4, \text{d}, +^5) \vdash (q_4, e, +^2) \vdash$
  (q_5, e, e)



PDA: Example 7

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^n\}$

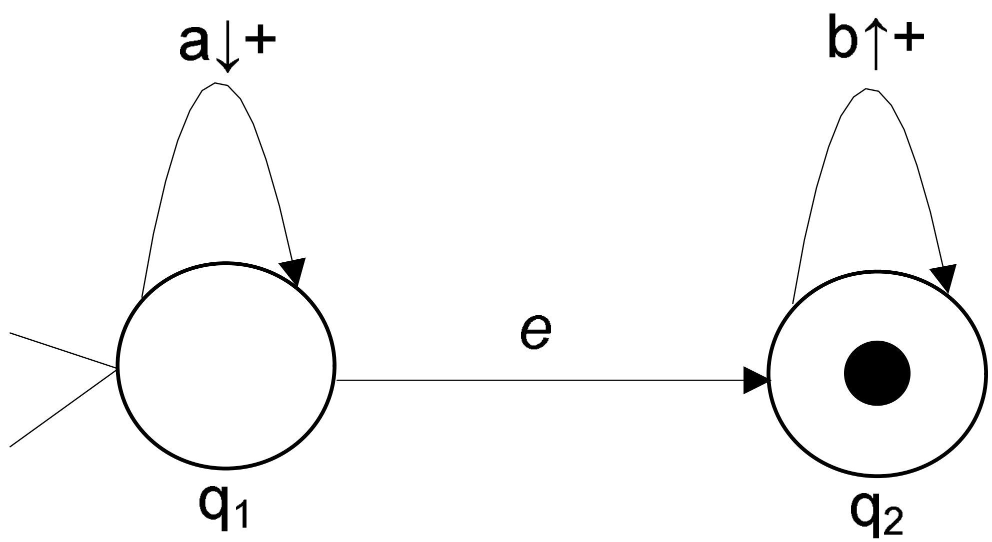
PDA: Example 7

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $a^n b^n$}

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

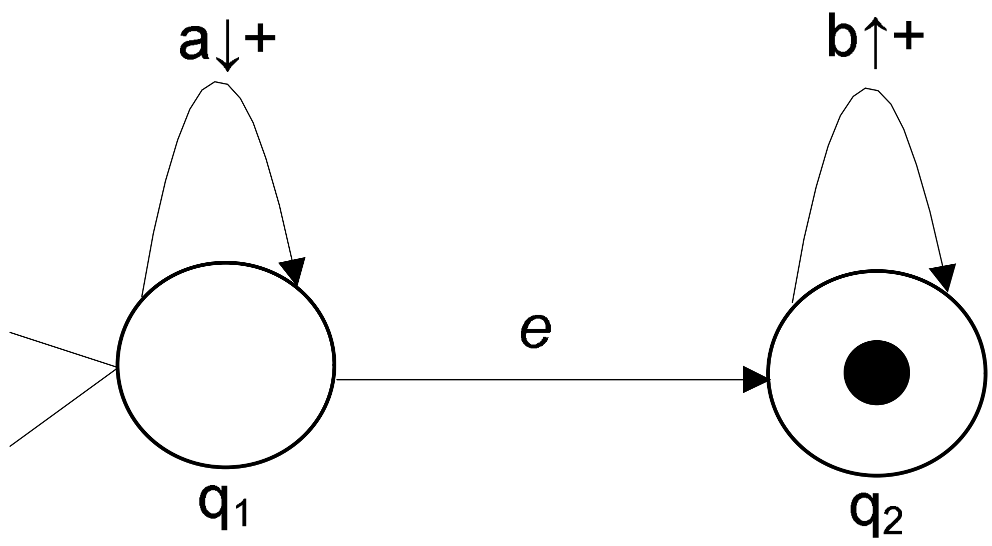
'+' — 1 "a" excess and 1 "b" lack



PDA: Example 7

- Computation for "aaaabbbb":

  (q_1, aaaabbbb, e) |- (q_1, aaabbbb, $+^1$) |-
  (q_1, aabbbb, $+^2$) |- (q_1, abbbb, $+^3$) |- (q_1, bbbb, $+^4$) |-
  (q_2, bbbb, $+^4$) |- (q_2, bbb, $+^3$) |- (q_2, bb, $+^2$) |-
  (q_2, b, +) |- (q_2, e, e)



PDA: Example 8

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{2n}\}$

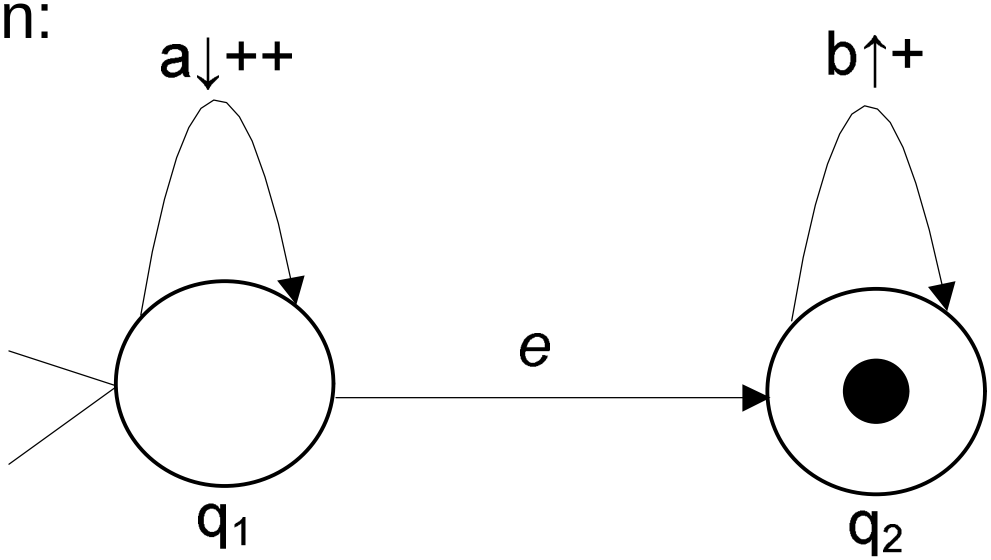
PDA: Example 8

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{2n}\}$
  - $L = \{w \in a^x b^y \mid 2x = y\}$

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1/2 "a" excess and 1 "b" lack

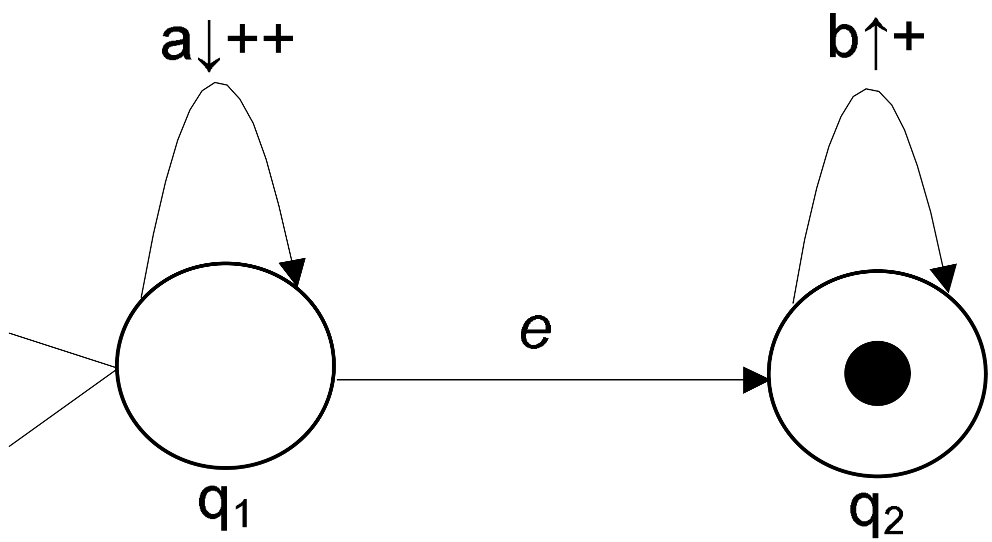
- Solution:



PDA: Example 8

- Computation for "aabbbb":

  $(q_1, \text{aabbbb}, e) \vdash (q_1, \text{abbbb}, +^2) \vdash (q_1, \text{bbbb}, +^4) \vdash$
  $(q_2, \text{bbbb}, +^4) \vdash (q_2, \text{bbb}, +^3) \vdash (q_2, \text{bb}, +^2) \vdash$
  $(q_2, \text{b}, +) \vdash (q_2, e, e)$



PDA: Example 9

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{3n}b^{2n}\}$

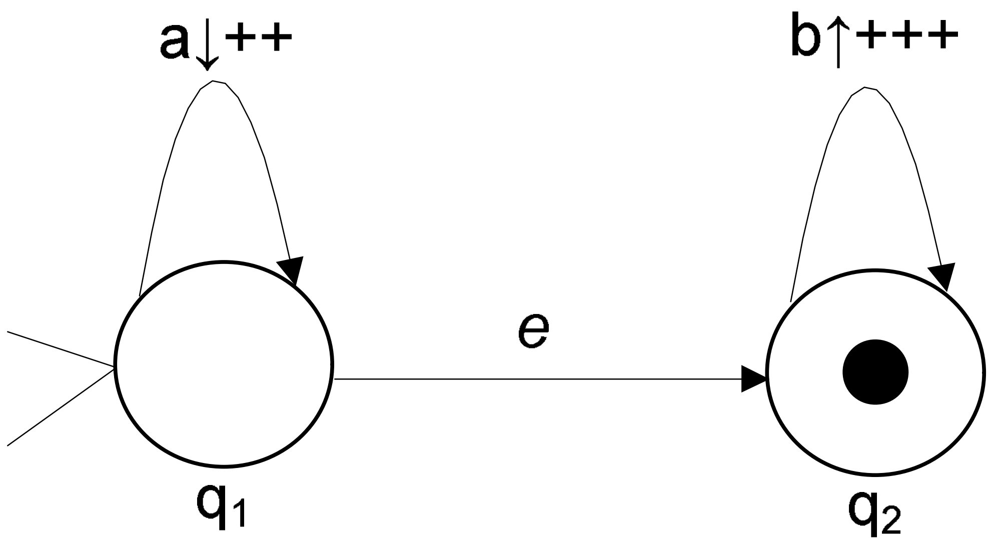
PDA: Example 9

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $a^{3n}b^{2n}$}
  - L = {w $\in$ $a^x b^y$ | 2x = 3y}

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1/2 "a" excess, 1/3 "b" lack

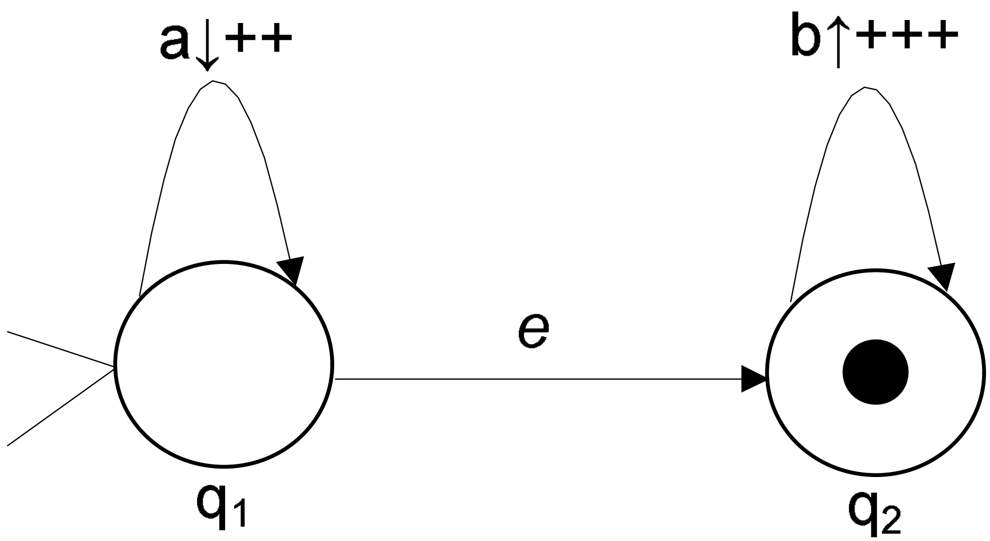
- Solution:



PDA: Example 9

- Computation for "aaabb":

  (q_1, aaabb, *e*) |- (q_1, aabb, $+^2$) |- (q_1, abb, $+^4$) |-
  (q_1, bb, $+^6$) |- (q_2, bb, $+^6$) |- (q_2, b, $+^3$) |- (q_2, *e*, *e*)



PDA: Example 10

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{3n+2}b^{2n}\}$
  - $L = \{w \in a^x b^y \mid 2(x - 2) = 3y\}$
    - 2 'a' will not change the stack
  - $L = \{w \in \{a, b\}^* \mid w = a^{3n}aab^{2n}\}$
    - nested problem:
      - inner problem: generate aa
      - outer problem: generate $a^{3n}b^{2n}$
  - $L = \{w \in a^x b^y \mid 2x = 3y + 4\}$
    - it would require the removal of $+^4$ at the end

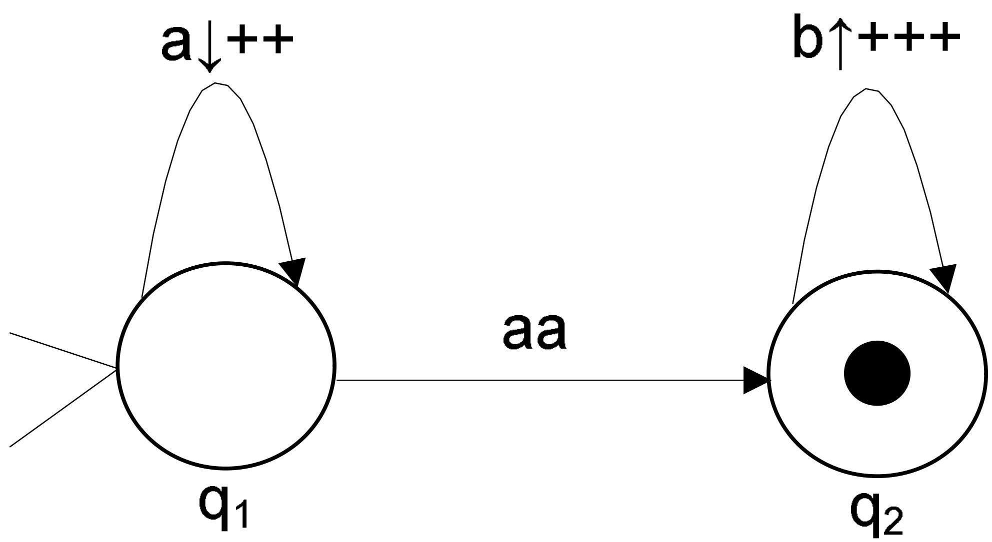
PDA: Example 10

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{3n+2}b^{2n}\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

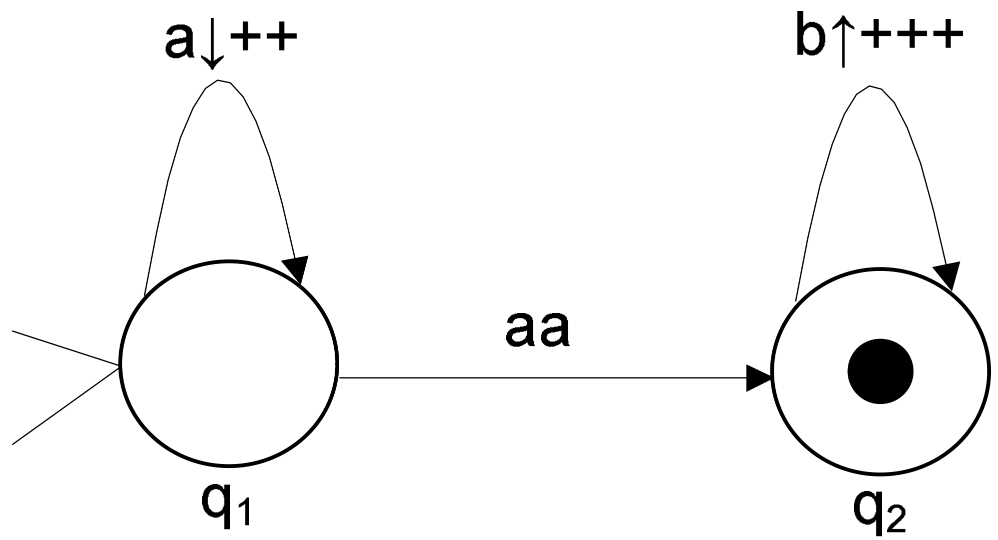
'+' — 1/2 "a" excess, 1/3 "b" lack



PDA: Example 10

- Computation for "aaaaabb":

  $(q_1, \text{aaaaabb}, e)$ |- $(q_1, \text{aaaabb}, +^2)$ |- $(q_1, \text{aaabb}, +^4)$ |- $(q_1, \text{aabb}, +^6)$ |- $(q_2, \text{bb}, +^6)$ |- $(q_2, \text{b}, +^3)$ |- (q_2, e, e)



PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

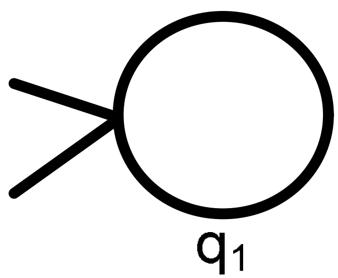
PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$
  - $L = \{w \in \{a, b\}^* \mid w = a^n bbb^n\}$

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack

- Solution:



PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack

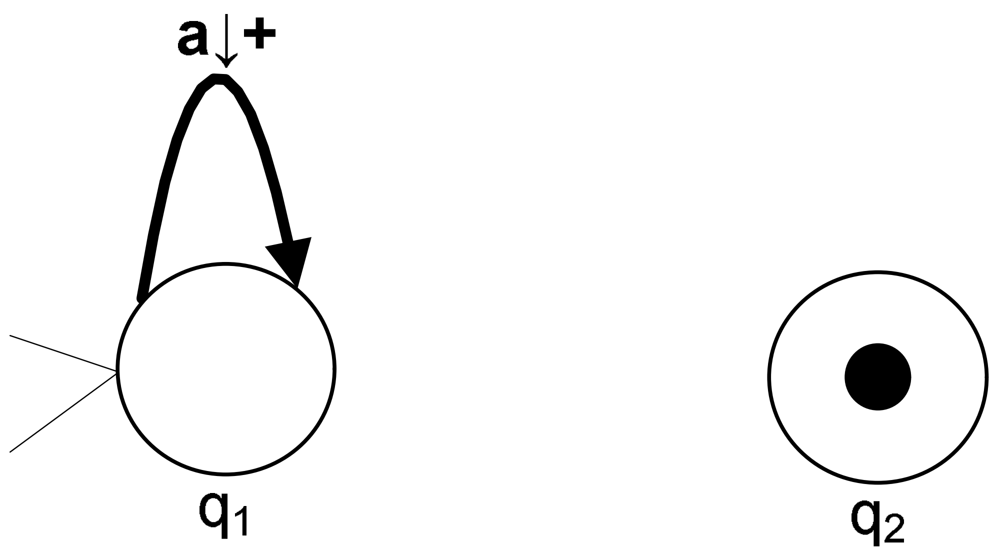
PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack



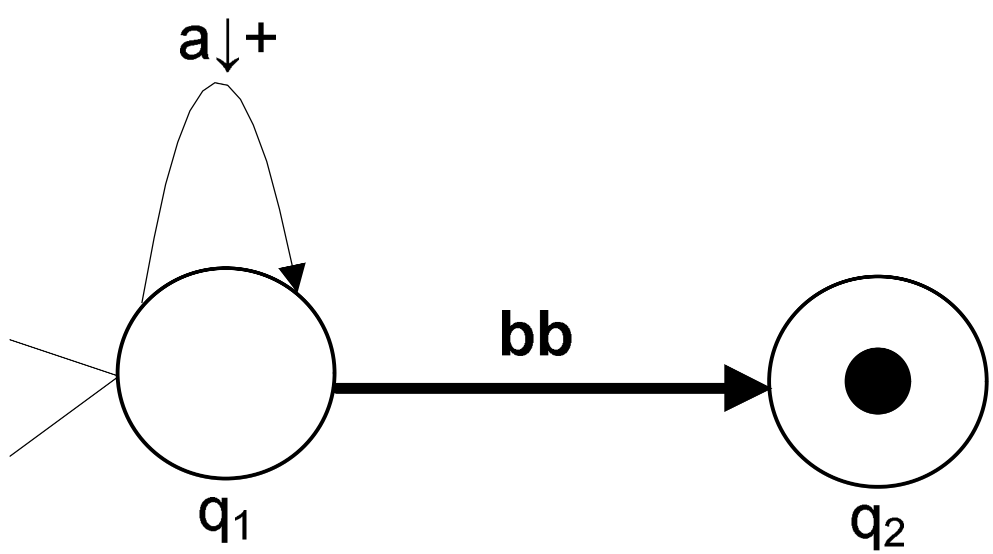
PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack



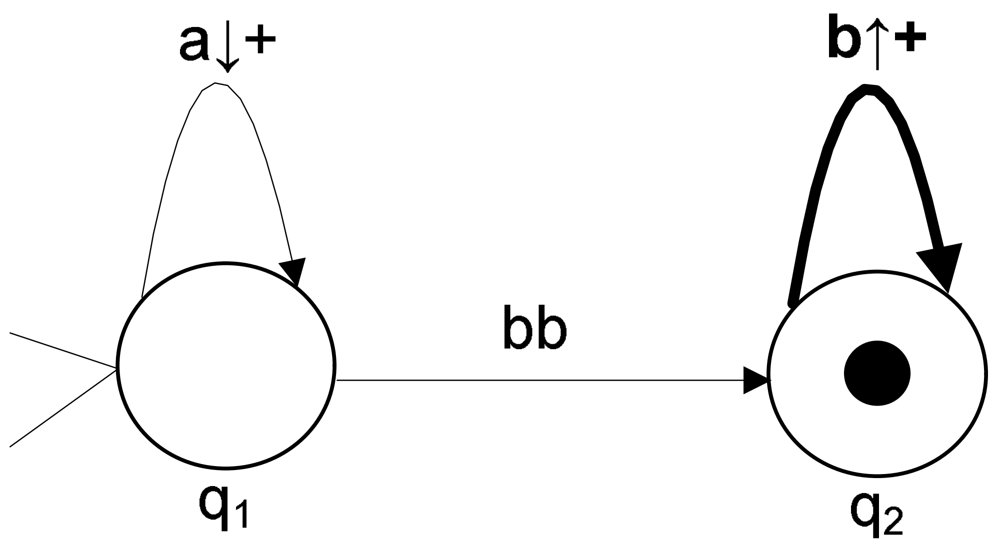
PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

- Solution:

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack



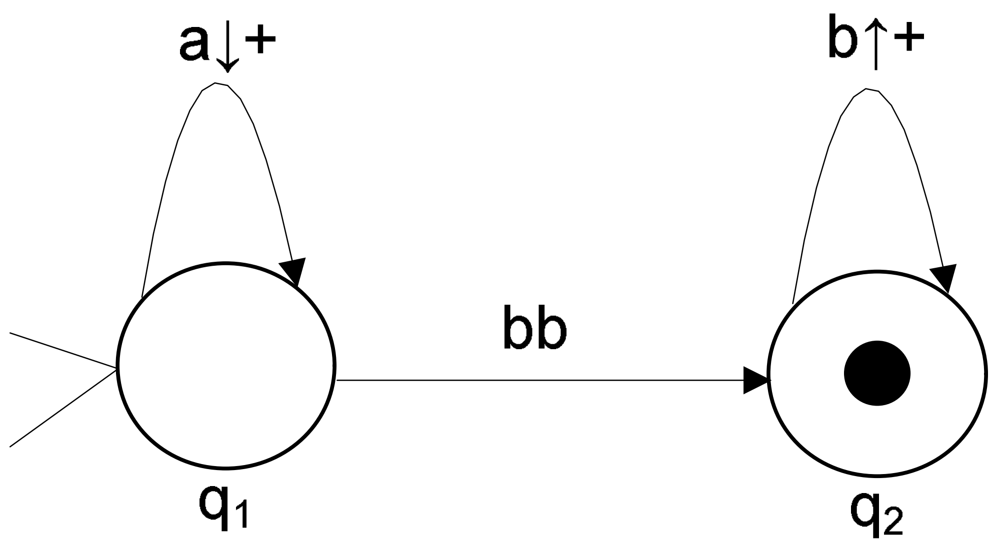
PDA: Example 11

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^n b^{n+2}\}$

- Solution:

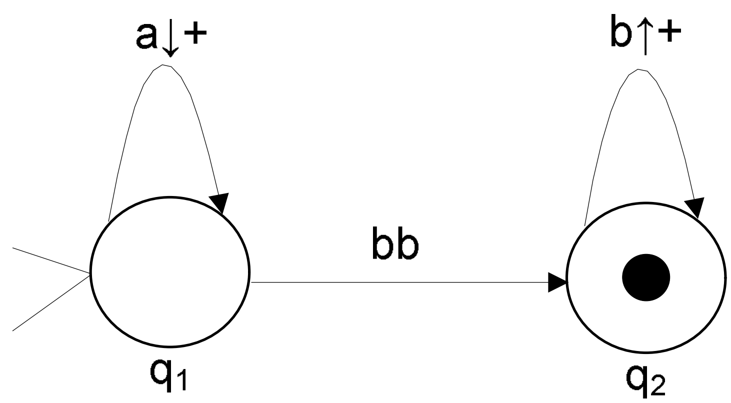
q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack



PDA: Example 11

State	**Unread input**	**Stack**
q_1	abbb	e
q_1	bbb	+
q_2	b	+
q_2	e	e



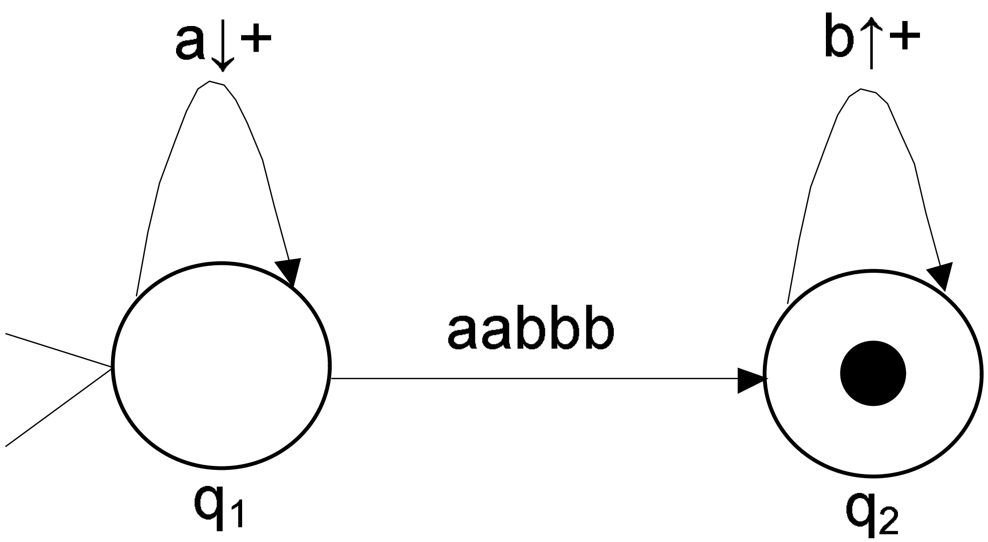
PDA: Example 12

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{n+2}b^{n+3}\}$
  - $L = \{w \in \{a, b\}^* \mid w = a^n aabbbb^n\}$

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 1 "a" excess, 1 "b" lack

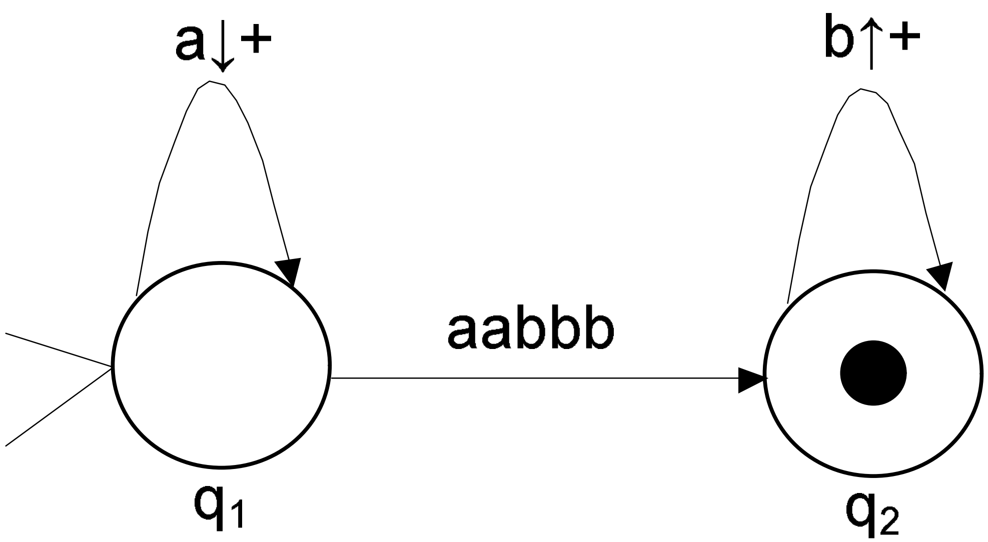
- Solution:



PDA: Example 12

- Computation for "aaaabbbbb":

  $(q_1$, aaaabbbbb, $e)$ |- $(q_1$, aaabbbbb, $+)$ |-
  $(q_1$, aabbbbb, $+^2)$ |- $(q_2$, bb, $+^2)$ |- $(q_2$, b, $+)$ |- (q_2, e, e)

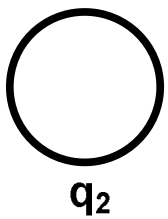


PDA: Example 13

- Give PDA M such that L(M) = L!
  - L = {w $\in$ $b^m a^m \cup a^{2m} b^m$, m ≥ 0}

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
  - $L = L_1 \cup L_2 = \{w \in b^m a^m, m \geq 0\} \cup \{a^{2m} b^m, m \geq 0\}$
    - using the construction theorem
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:

q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

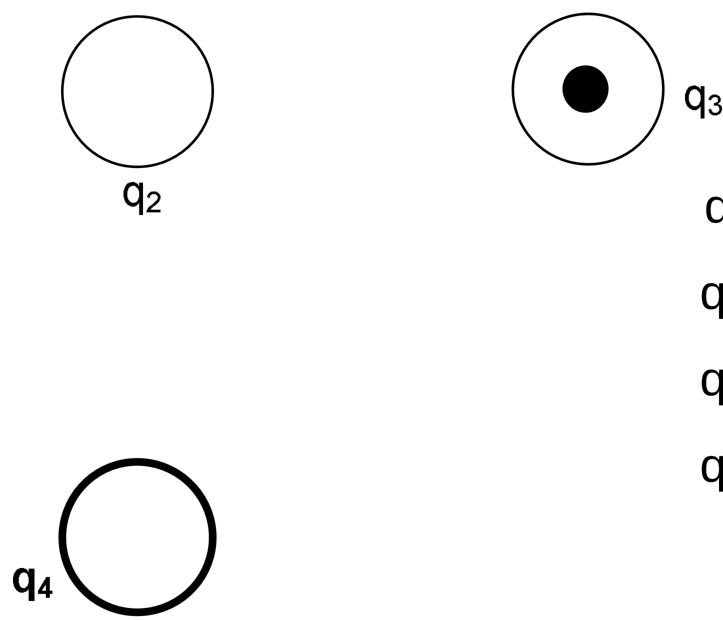
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

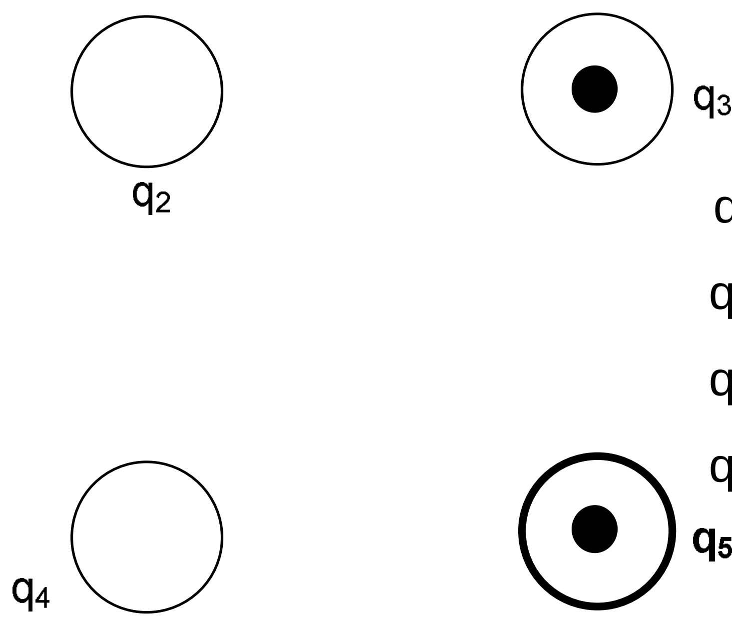
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

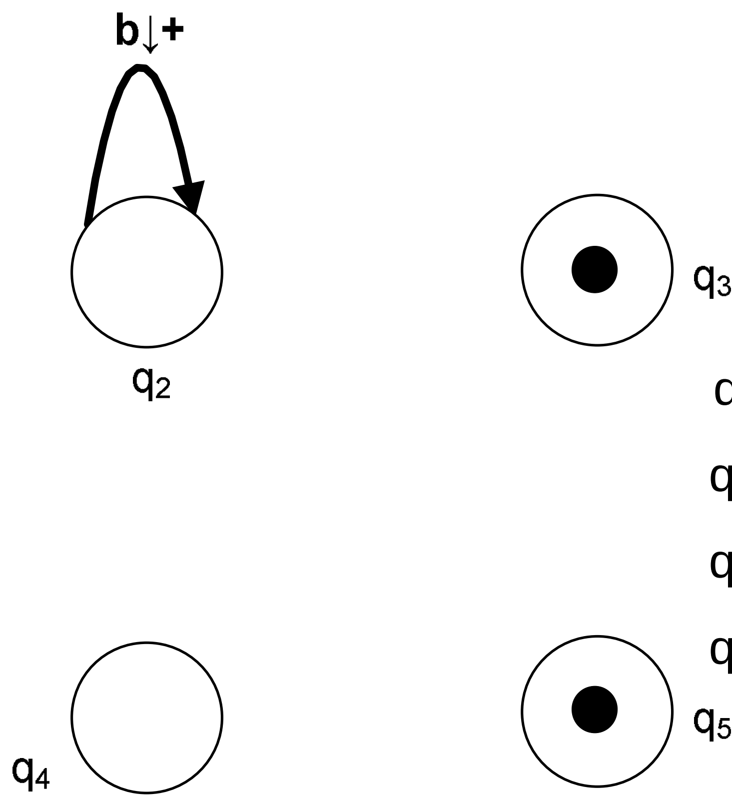
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

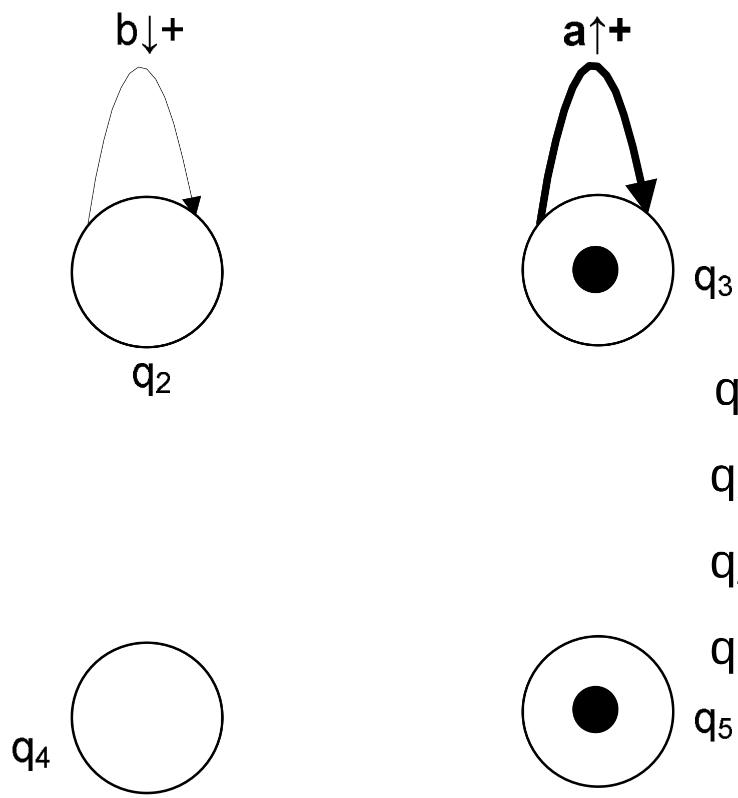
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - L = $\{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

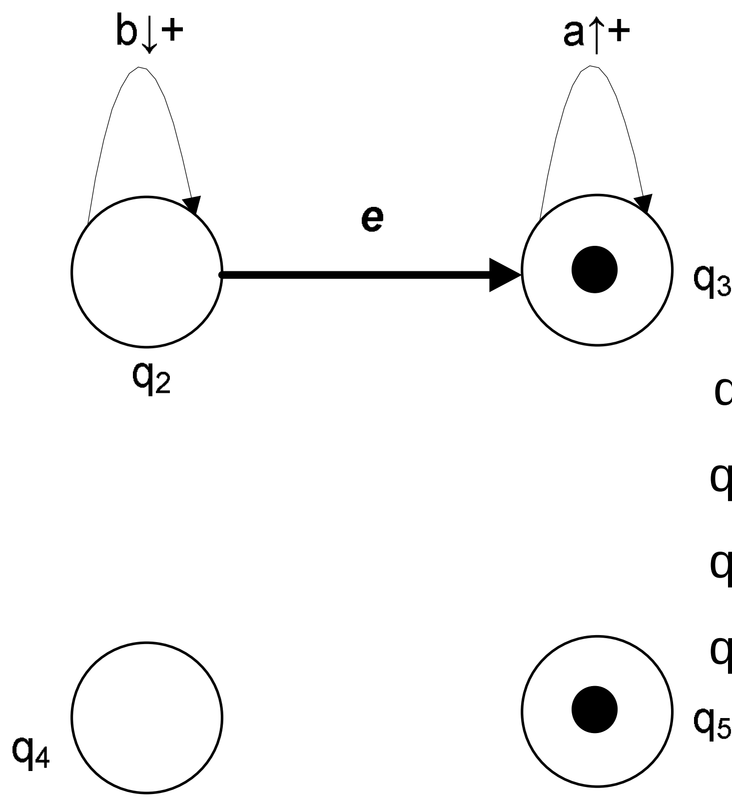
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

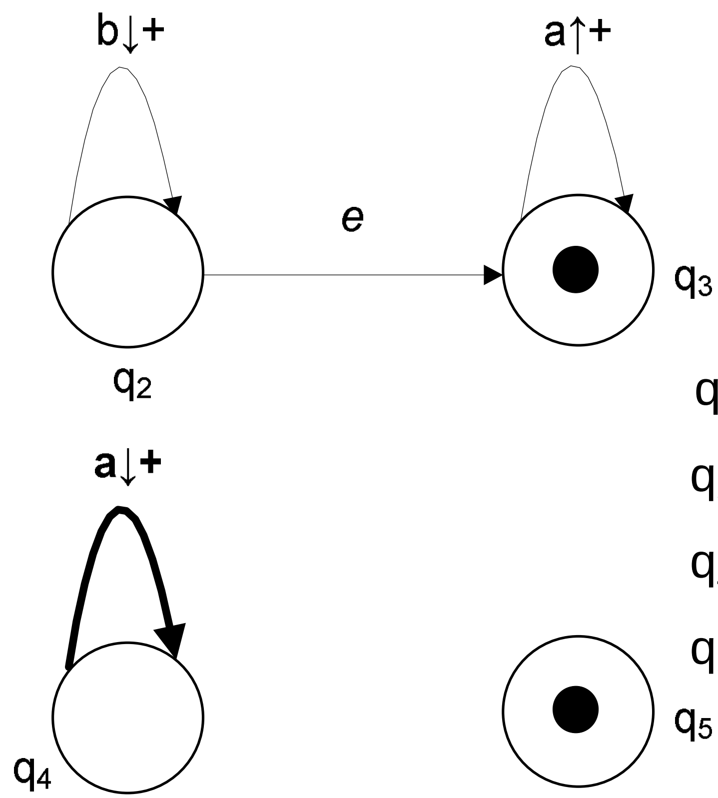
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

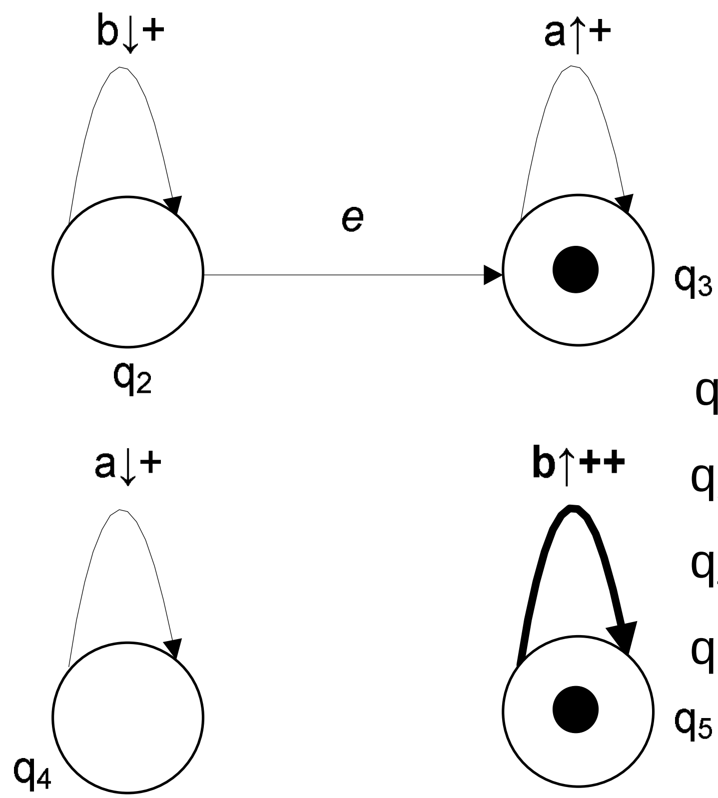
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

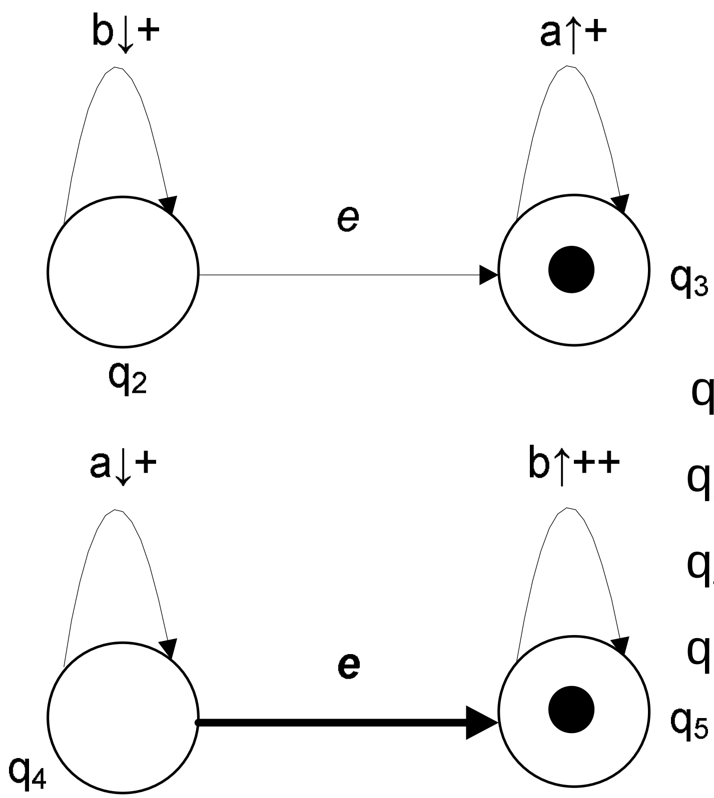
q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$

- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

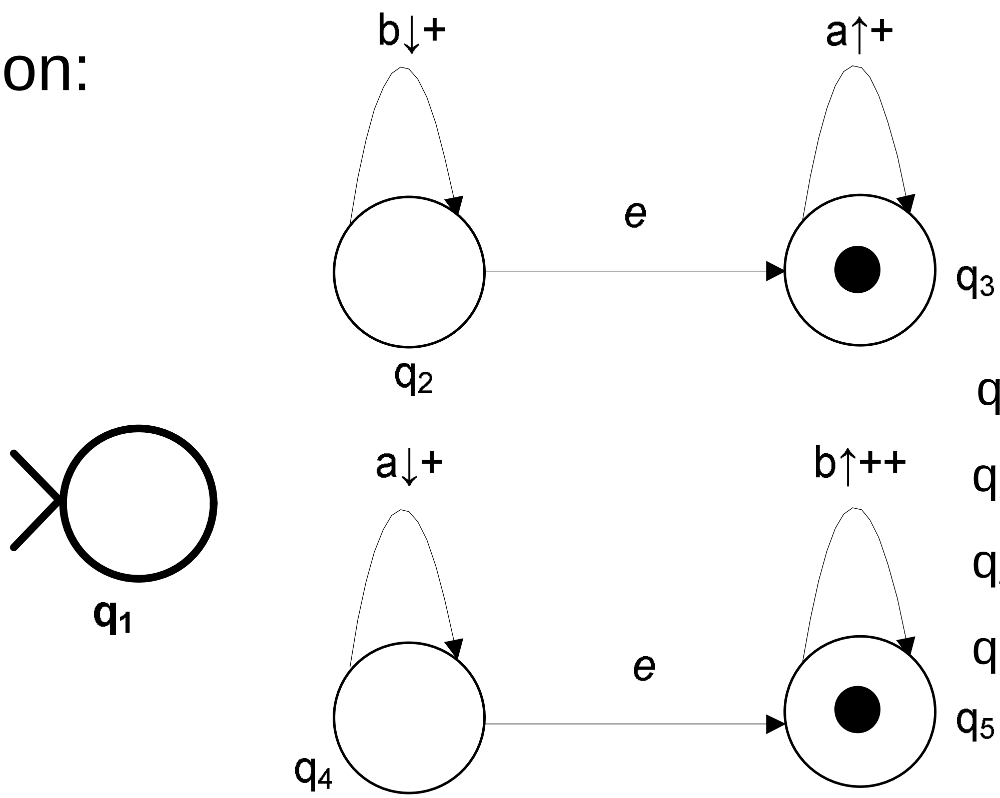
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

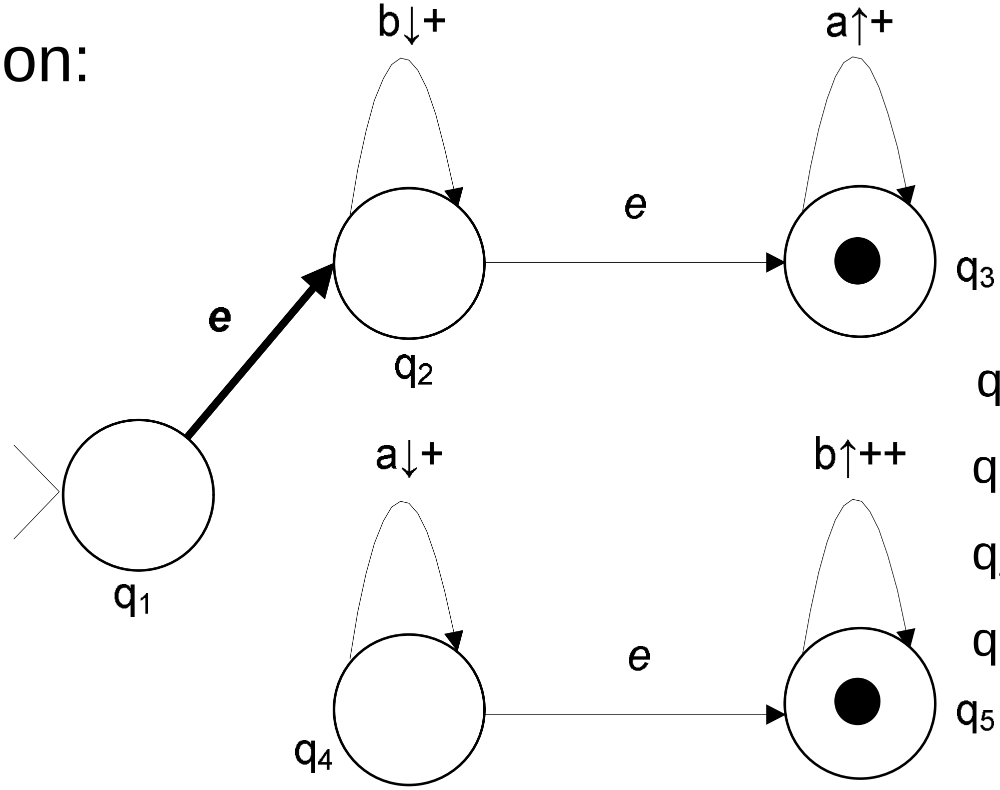
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - L = $\{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

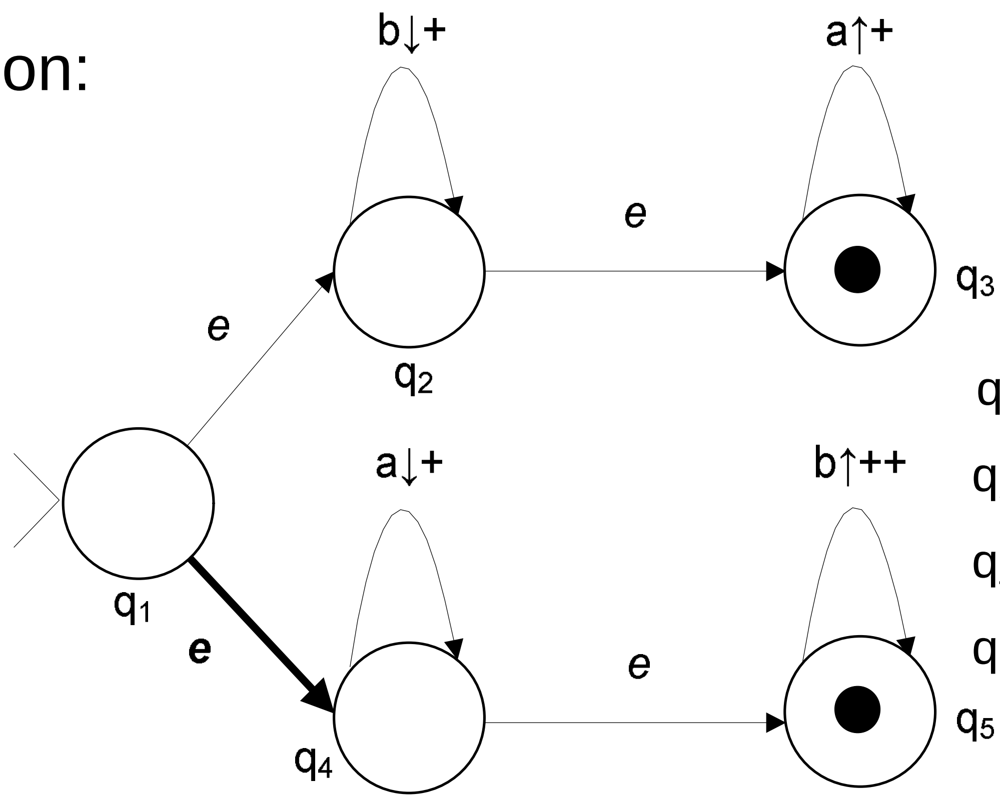
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

q_2 — 'b' reader state for L_1

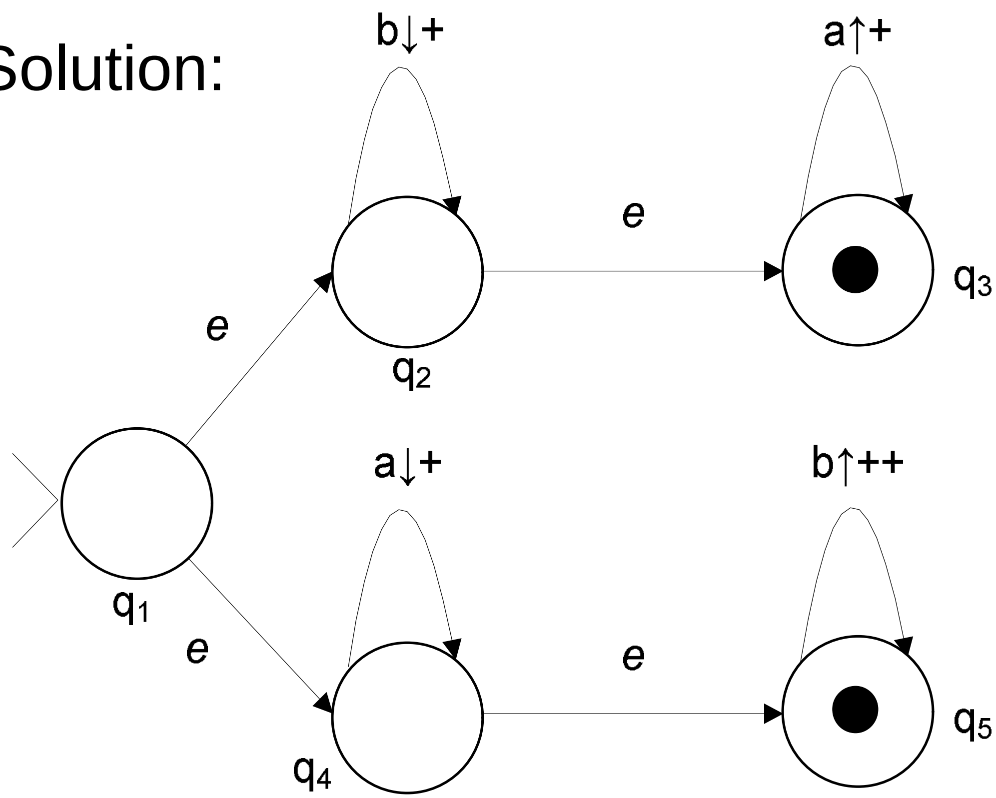
q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

PDA: Example 13

- Give PDA M such that L(M) = L!
  - $L = \{w \in b^m a^m \cup a^{2m} b^m, m \geq 0\}$
- Solution:



q_1 — initial state

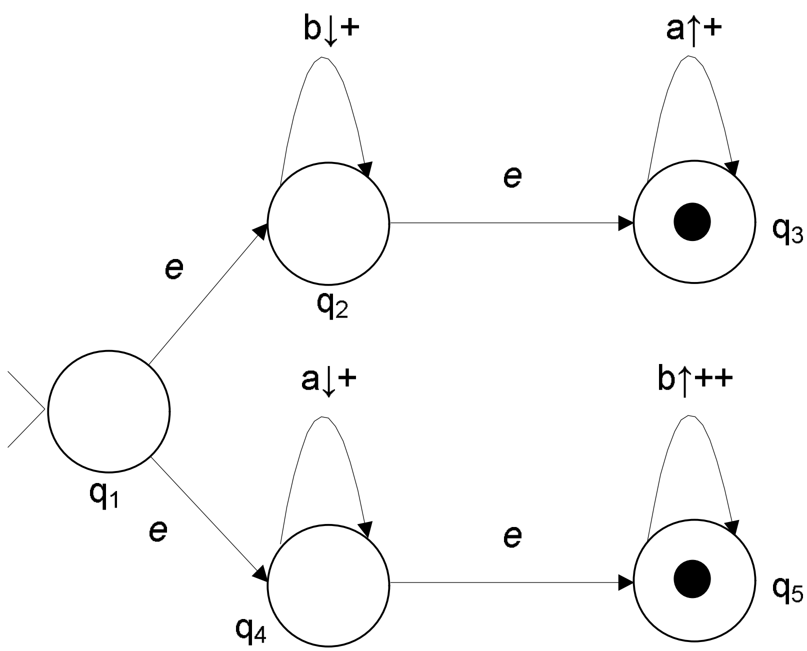
q_2 — 'b' reader state for L_1

q_3 — 'a' reader state for L_1

q_4 — 'a' reader state for L_2

q_5 — 'b' reader state for L_2

State	Unread input	Stack
q_1	bbaa	e
q_2	bbaa	e
q_2	baa	+
q_2	aa	++
q_3	aa	++
q_3	a	+
q_3	e	e



PDA: Example 14

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $a^{2n}uu^Rb^{3n}$}
    - nested problem

PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

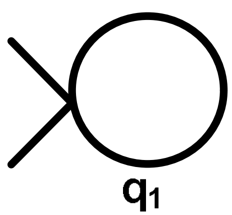
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - L = $\{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u

PDA: Example 14

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $a^{2n}uu^Rb^{3n}$}

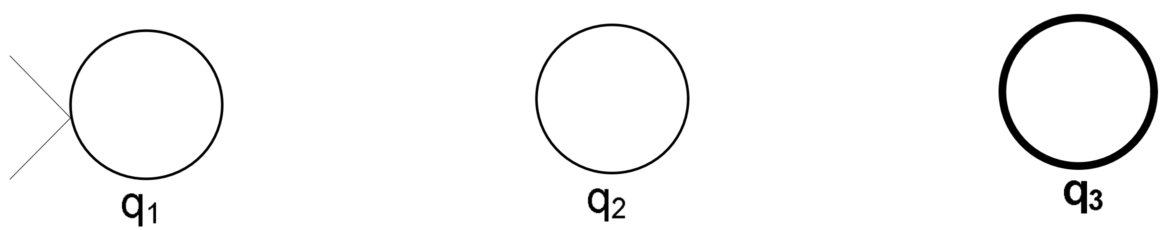
- Solution:

q_1 — 'a' reader state | q_2 — reading u

q_4 — 'b' reader state | q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u

PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$
- Solution:

q_1 — 'a' reader state

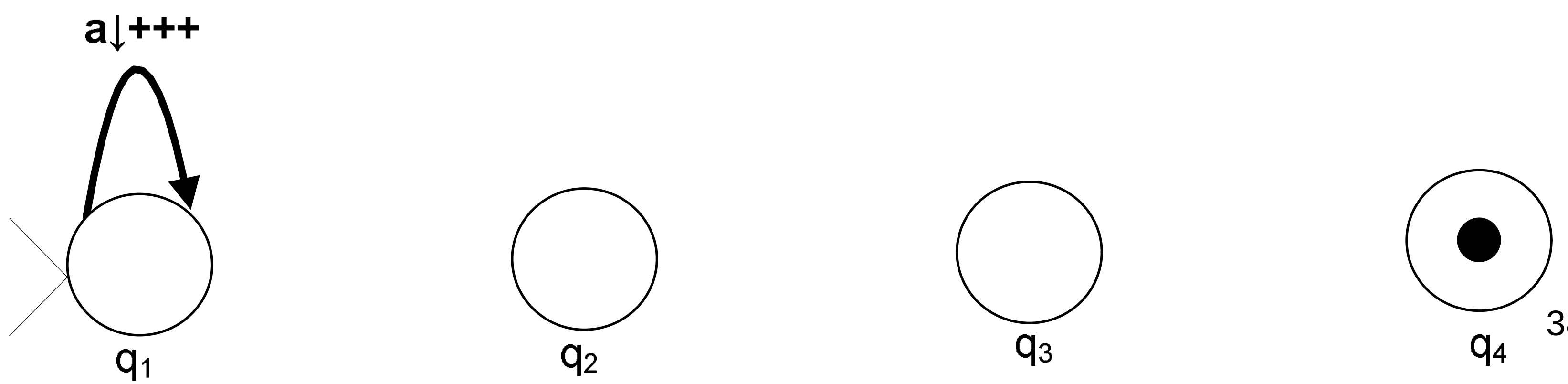
q_2 — reading u

q_4 — 'b' reader state

q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

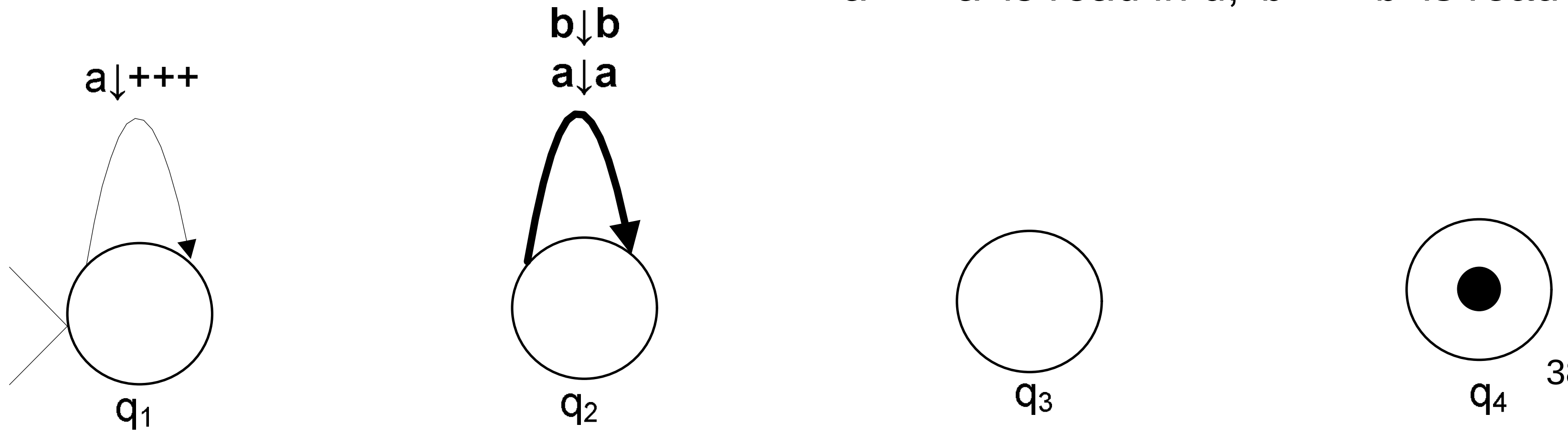
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

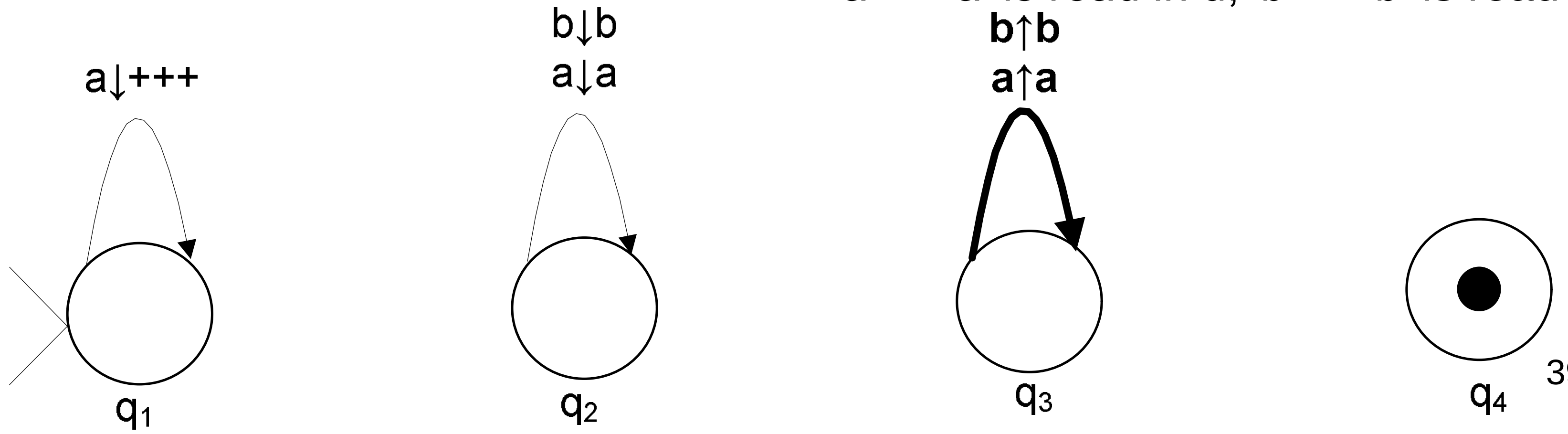
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

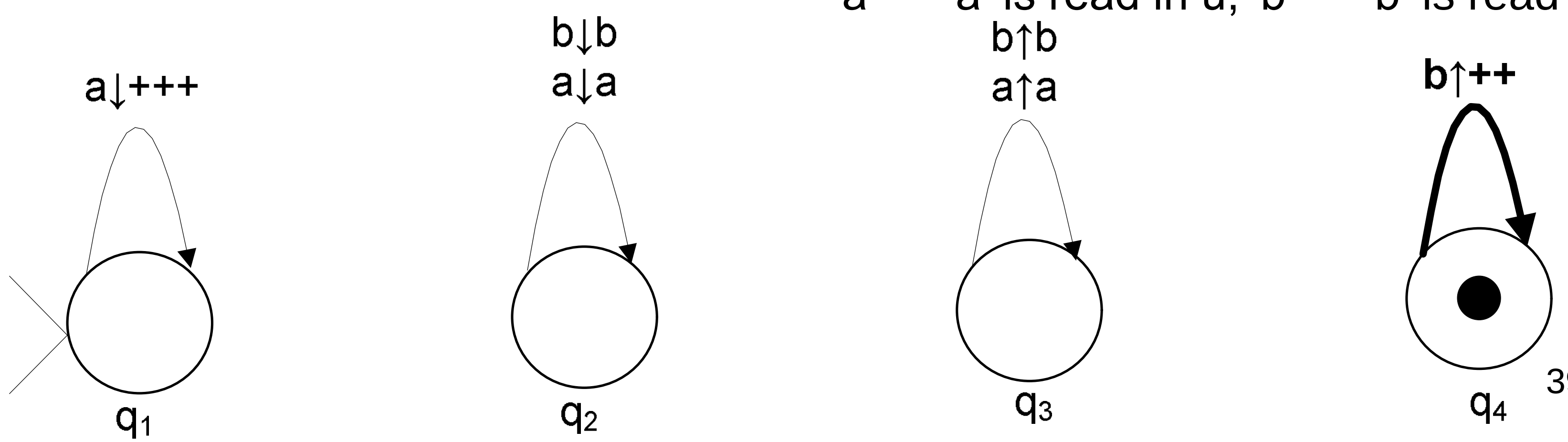
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $a^{2n}uu^Rb^{3n}$}

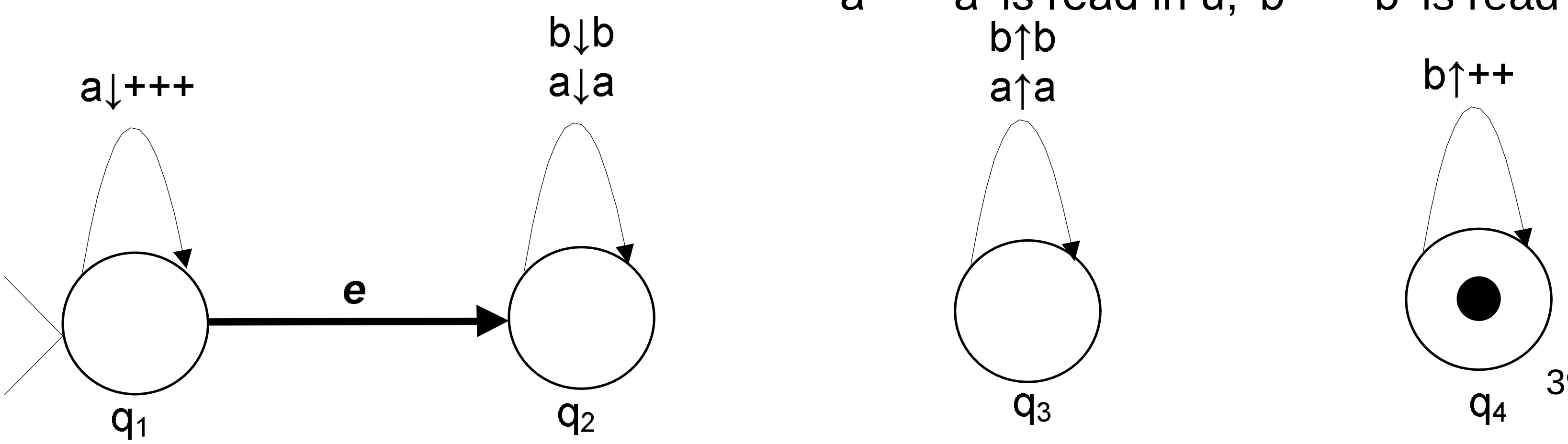
- Solution:

q_1 — 'a' reader state q_2 — reading u

q_4 — 'b' reader state q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - – $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

- Solution:

q_1 — 'a' reader state

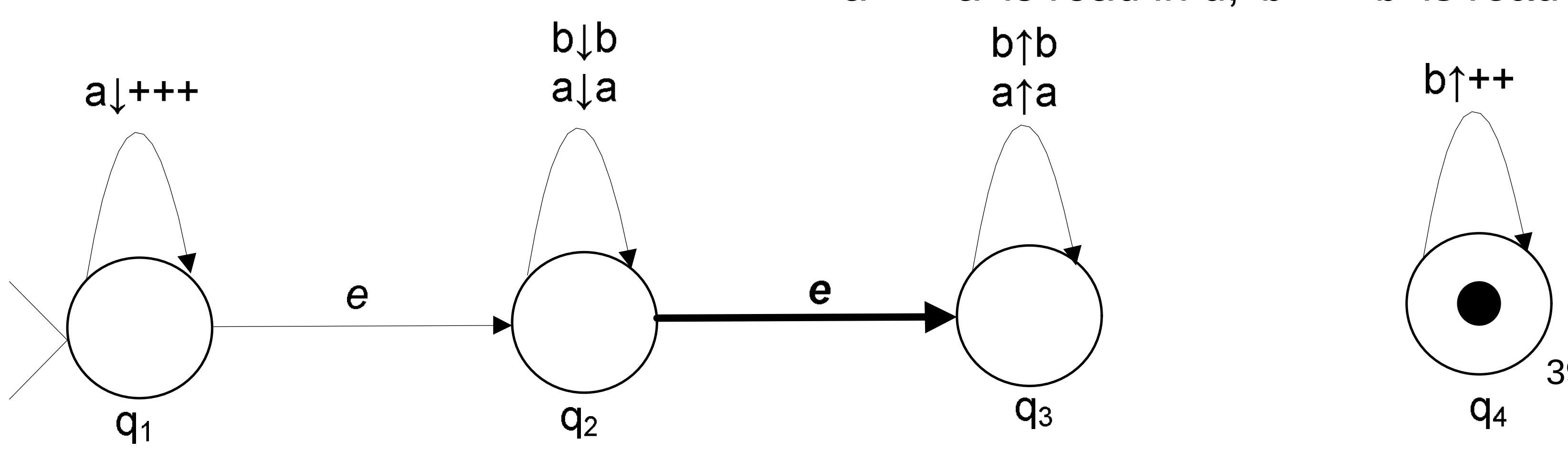
q_2 — reading u

q_4 — 'b' reader state

q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u

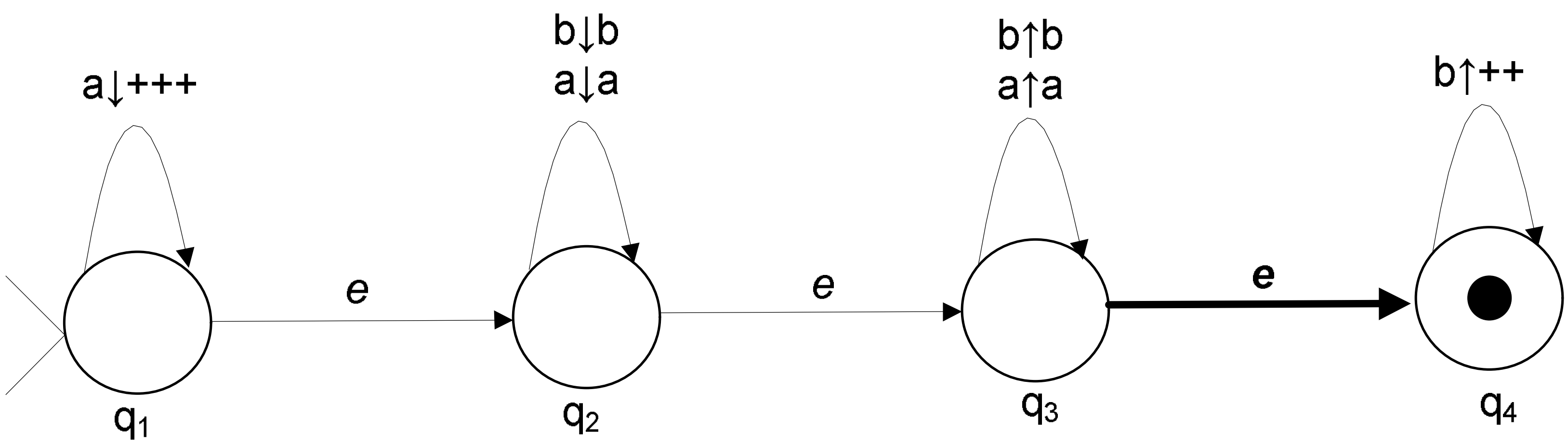


PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

- Solution:

q_1 — 'a' reader state
q_2 — reading u
q_4 — 'b' reader state
q_3 — reading u^R
'+' — 1/3 'a' excess, 1/2 'b' lack
'a' — 'a' is read in u, 'b' — 'b' is read in u



PDA: Example 14

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{2n}uu^Rb^{3n}\}$

- Solution:

q_1 — 'a' reader state

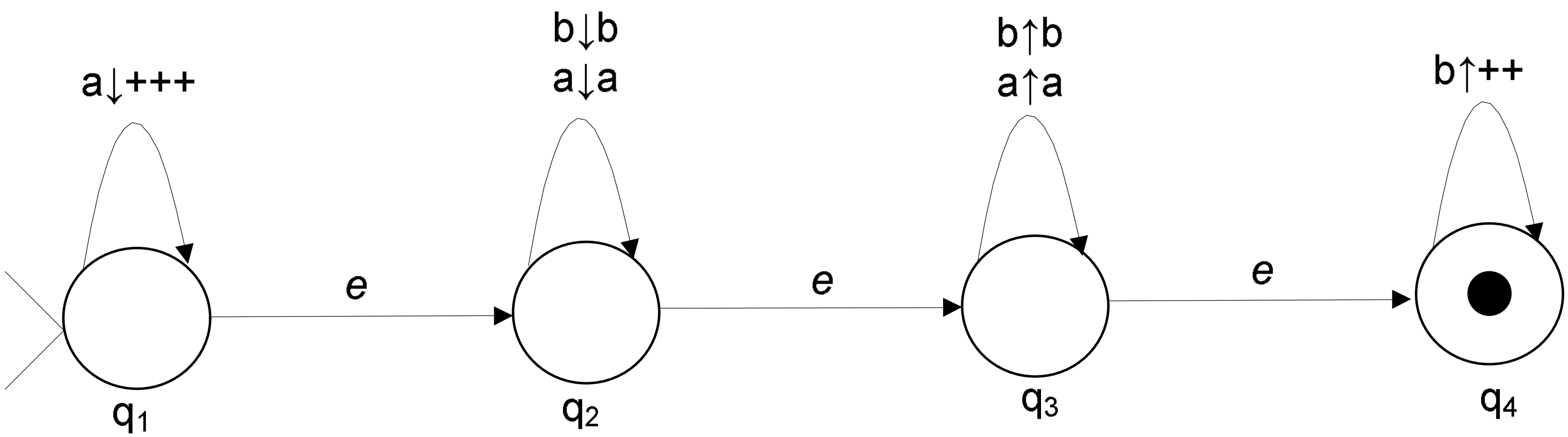
q_2 — reading u

q_4 — 'b' reader state

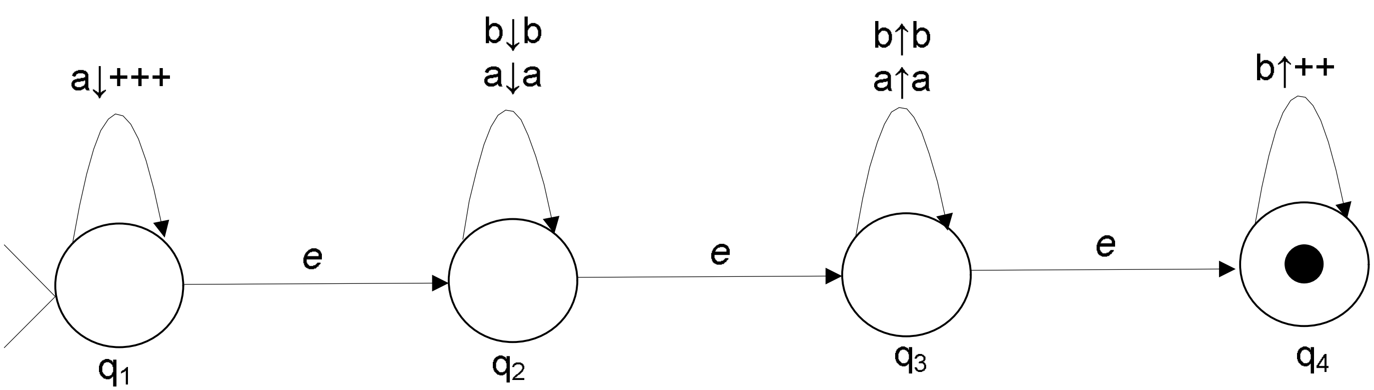
q_3 — reading u^R

'+' — 1/3 'a' excess, 1/2 'b' lack

'a' — 'a' is read in u, 'b' — 'b' is read in u



State	Unread input	Stack
q_1	aaabbabbb	e
q_1	aabbabbb	$+^3$
q_1	abbabbb	$+^6$
q_2	abbabbb	$+^6$
q_2	bbabbb	$a+^6$
q_2	babbb	$ba+^6$
q_3	babbb	$ba+^6$
q_3	abbb	$a+^6$
q_3	bbb	$+^6$
q_4	bbb	$+^6$
q_4	bb	$+^4$
q_4	b	$+^2$
q_4	e	e



PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$

PDA: Example 15

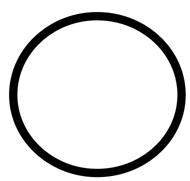
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$
  - $L = \{w \in a^{2n}b^{n+2}\}^*$
    - using the construction theorem
    - '+' has new interpretation now
- Solution:

q_1 — start state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



$\mathbf{q_2}$

PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$
- Solution:

q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack

PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$

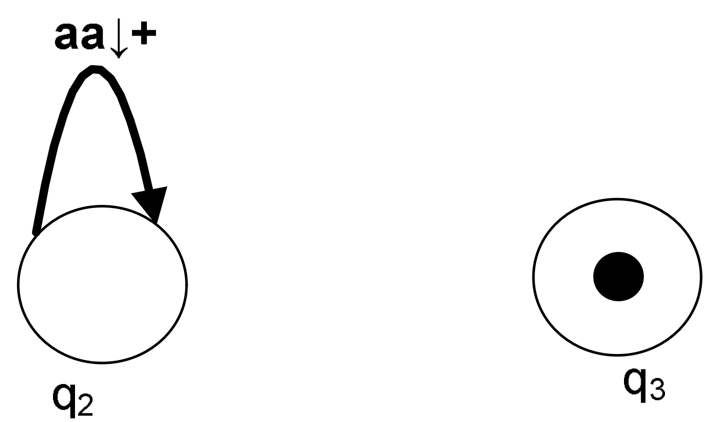
- Solution:

q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$

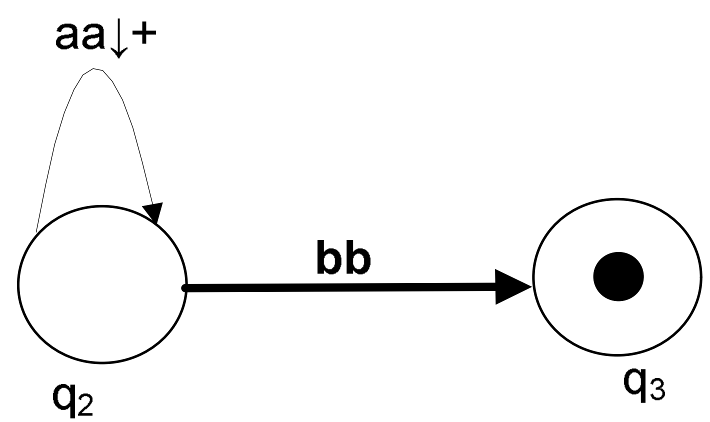
q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack

- Solution:



PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$

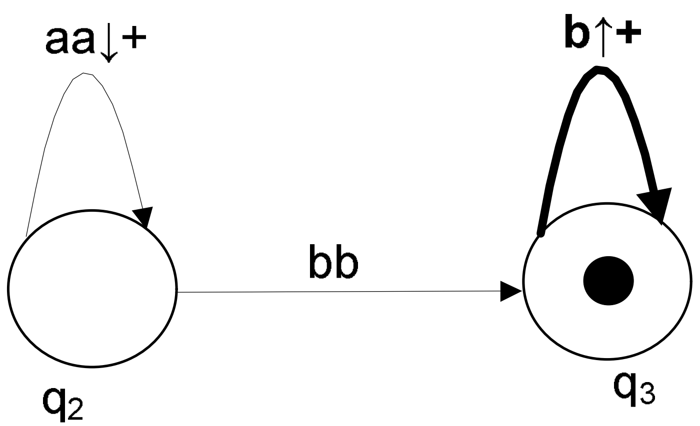
q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack

- Solution:



PDA: Example 15

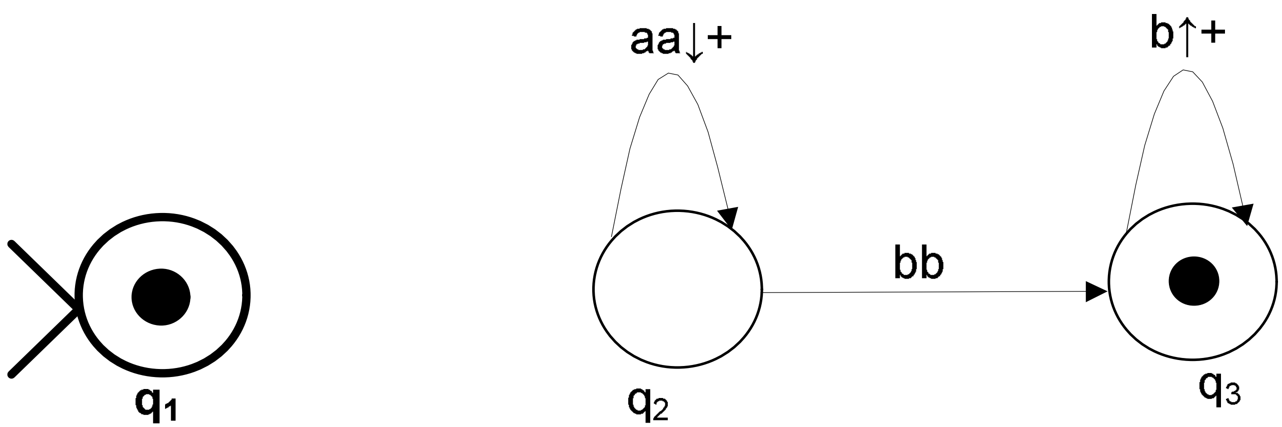
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$
- Solution:

q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



PDA: Example 15

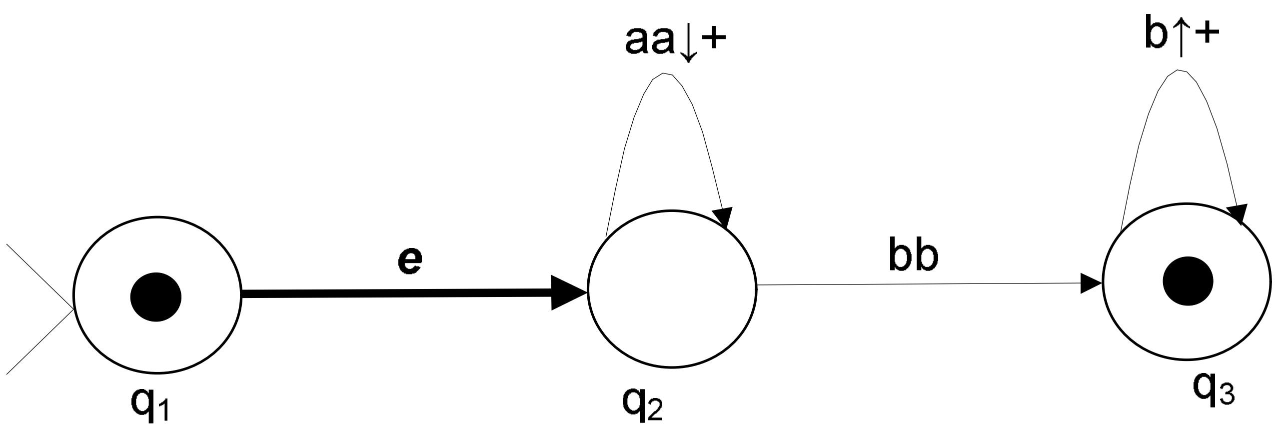
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$
- Solution:

q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



PDA: Example 15

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$

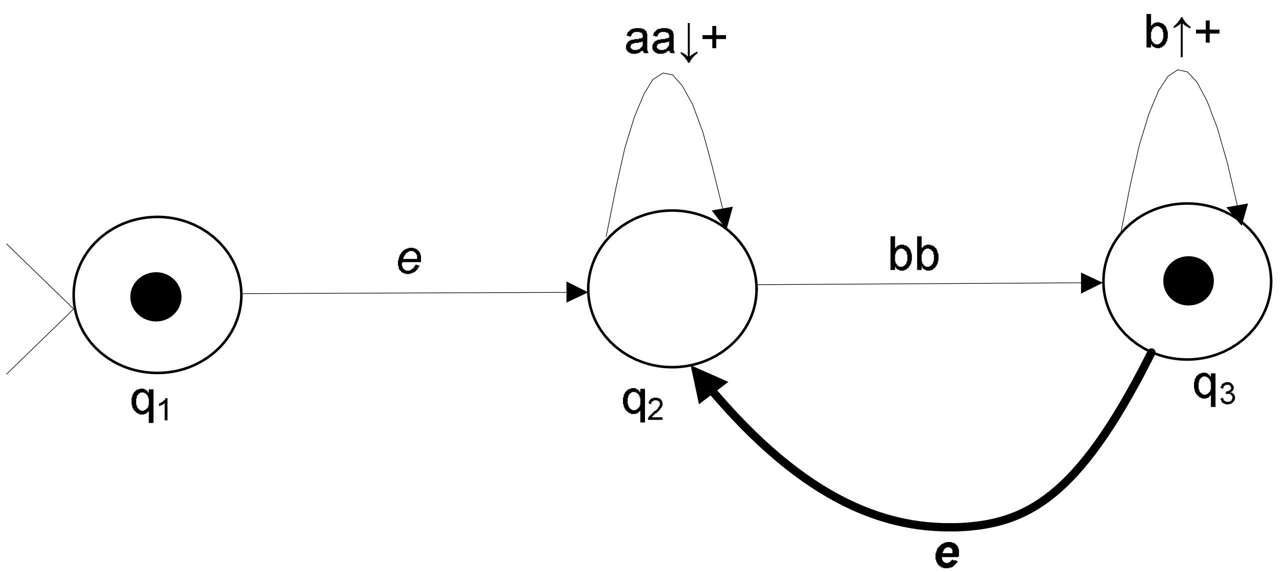
- Solution:

q_1 — initial state

q_2 — 'a' reader state

q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



PDA: Example 15

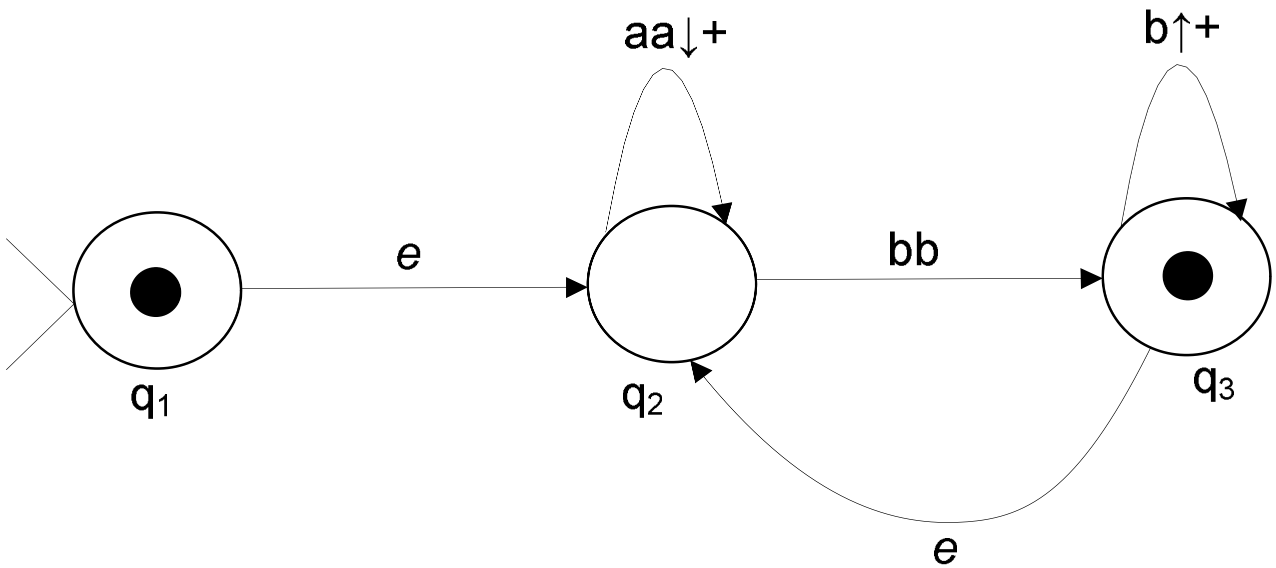
- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = (a^{2n}b^{n+2})^*\}$
- Solution:

q_1 — initial state

q_2 — 'a' reader state

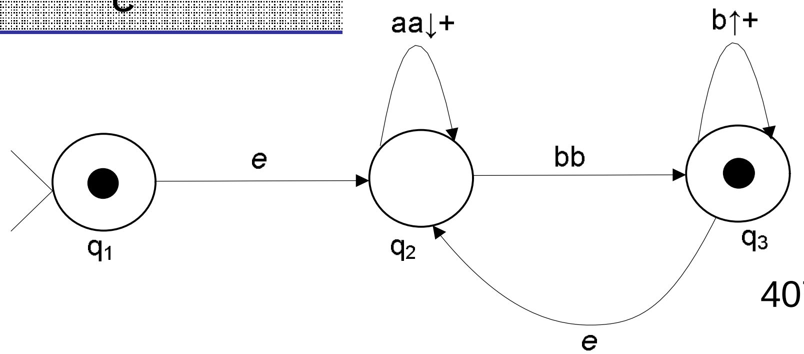
q_3 — 'b' reader state

'+' — 2 'a' excess
1 'b' lack



PDA: Example 15

State	Unread input	Stack
q_1	aabbbaabbb	*e*
q_2	aabbbaabbb	*e*
q_2	bbbaabbb	+
q_3	baabbb	+
q_3	aabbb	*e*
q_2	aabbb	*e*
q_2	bbb	+
q_3	b	+
q_3	*e*	*e*



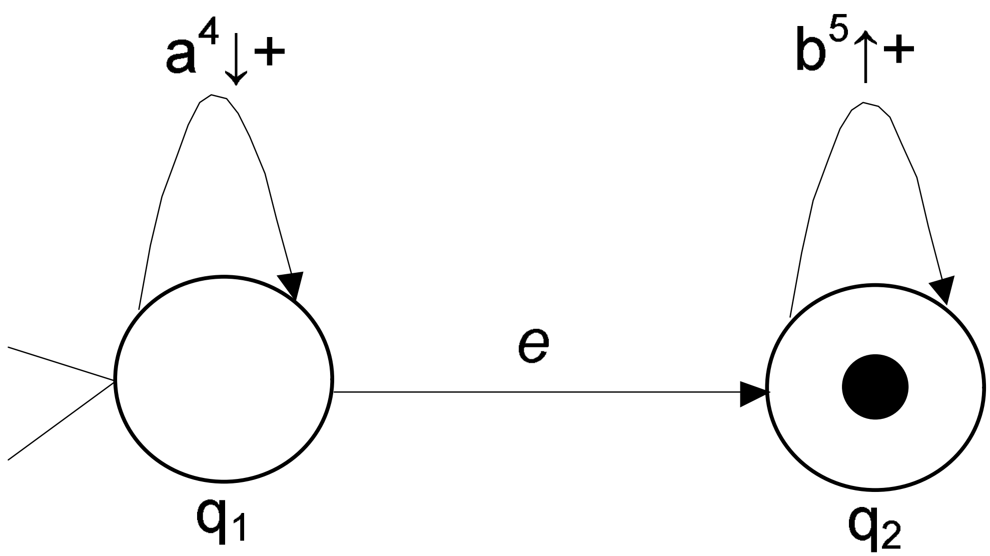
PDA: Example 16

- Give PDA M such that L(M) = L!
  - $L = \{w \in \{a, b\}^* \mid w = a^{4n}b^{5n}\}$

q_1 — 'a' reader state q_2 — 'b' reader state

'+' — 4 "a" excess, 5 "b" lack

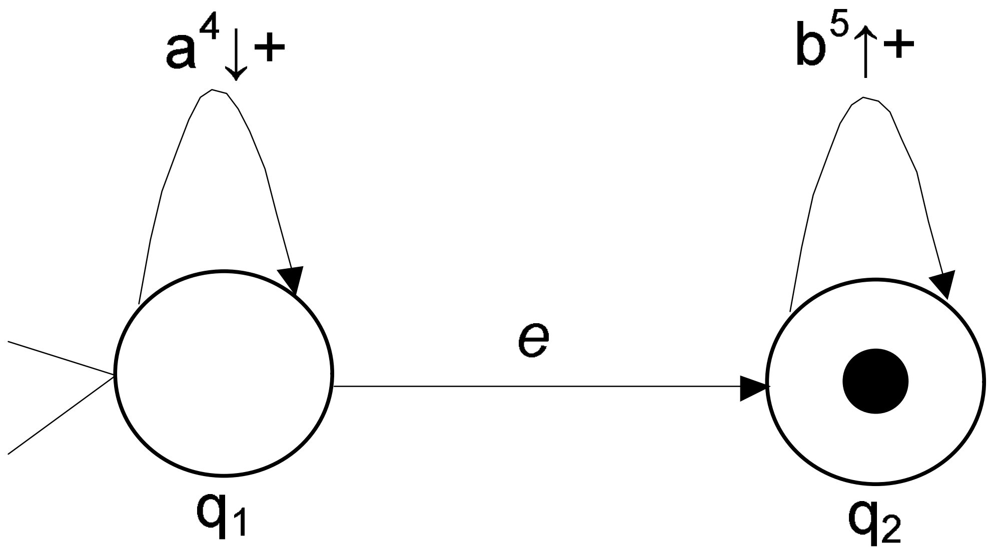
- Solution:



PDA: Example 16

- Computation for "a^8b^{10}":

  (q_1, a^8b^{10}, e) |- $(q_1, a^4b^{10}, +)$ |- $(q_1, b^{10}, ++)$ |-
  $(q_2, b^{10}, ++)$ |- $(q_2, b^5, +)$ |- (q_2, e, e)



PDA: Example 17

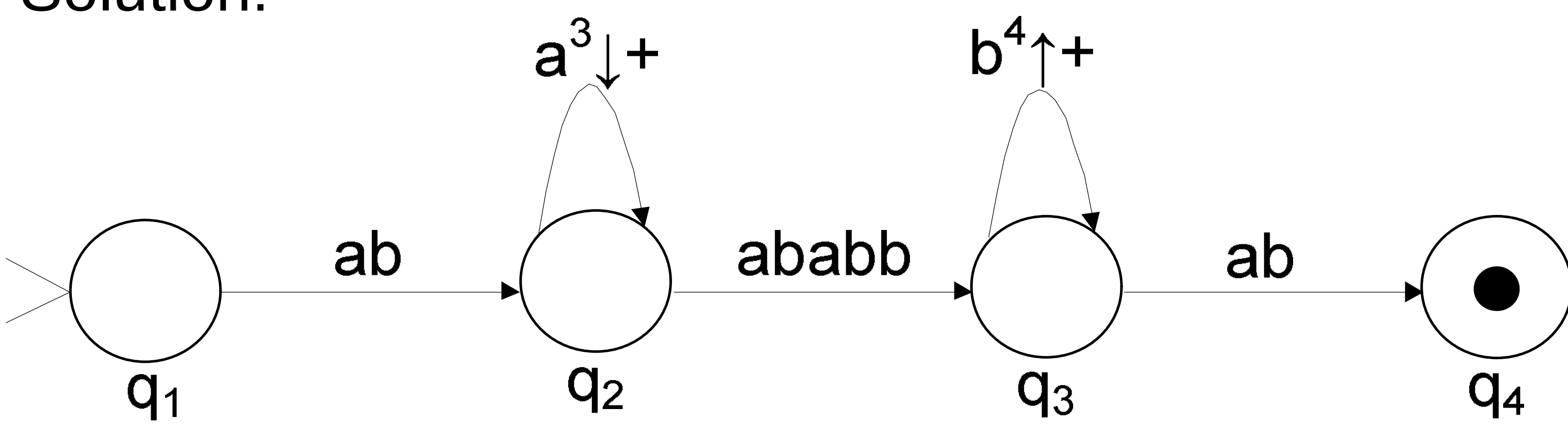
- Give PDA M such that L(M) = L!
  - L = {w $\in$ {a, b}* | w = $aba^{3n+1}bab^{4n+2}ab$}
  - L = {w $\in$ {a, b}* | w = $aba^{3n}ababbb^{4n}ab$}

q_1 — initial state q_2 — 'a' reader state

q_3 — 'b' reader state q_2 — final state

'+' — 3 "a" excess, 4 "b" lack

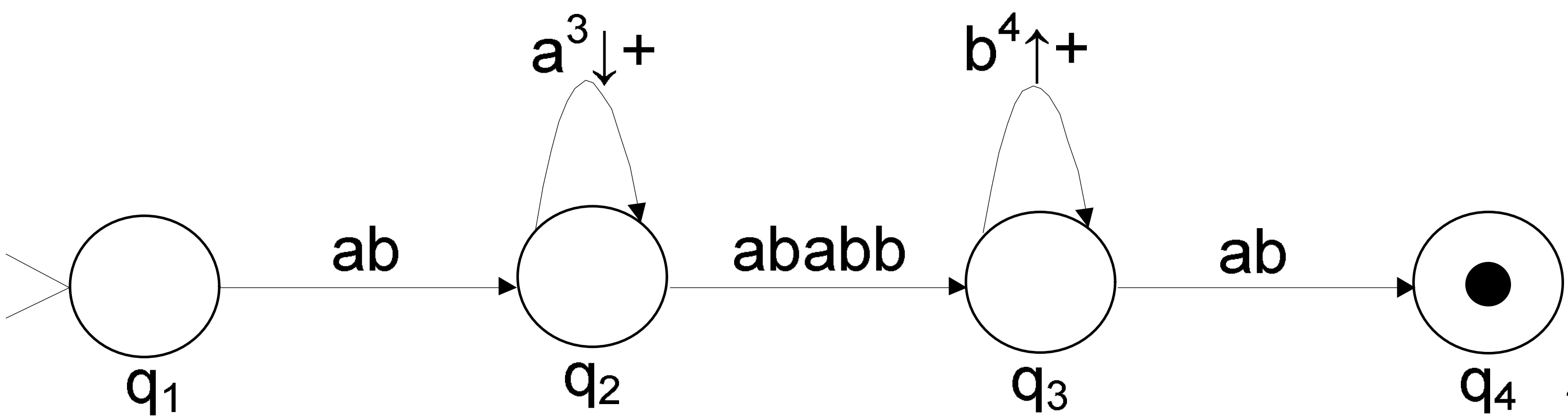
- Solution:



PDA: Example 17

- Computation for "abaaaababbbbbbab":

  (q_1, abaaaababbbbbbab, e) |-
  (q_2, aaaababbbbbbab, e) |- (q_2, ababbbbbbab, +) |-
  (q_3, bbbbab, +) |- (q_3, ab, +) |- (q_4, e, e)



Element of the Theory of the Computation

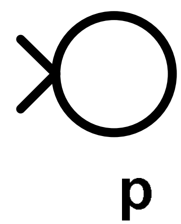
Lecture 9

CFG → PDA: Example 1

- Construct PDA M such that L(M) = L(G)!
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aSb | bSa | *e*}

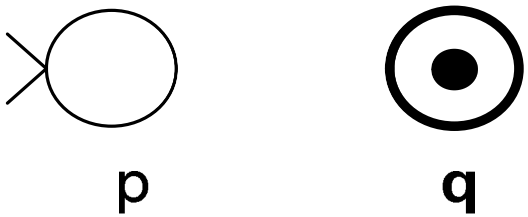
CFG → PDA: Example 1

- Step a1: introduce an inital state
  - $V = \{a, b, S\}$
  - $\Sigma = \{a, b\}$
  - $S = S$
  - $R = \{S \rightarrow aSb \mid bSa \mid e\}$



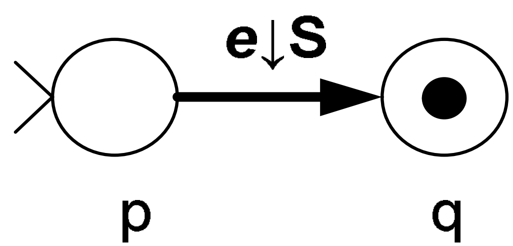
CFG → PDA: Example 1

- Step a2: introduce a final state
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aSb | bSa | *e*}



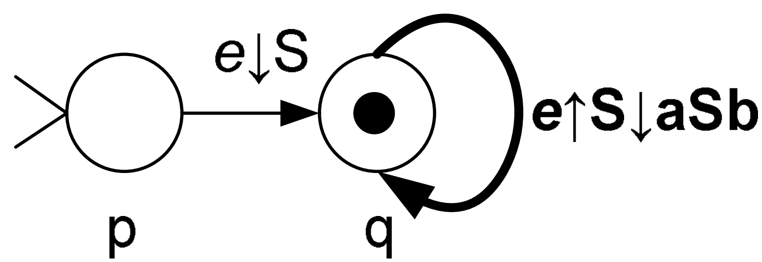
CFG → PDA: Example 1

- Step a3: initially push S into the stack
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aSb | bSa | *e*}



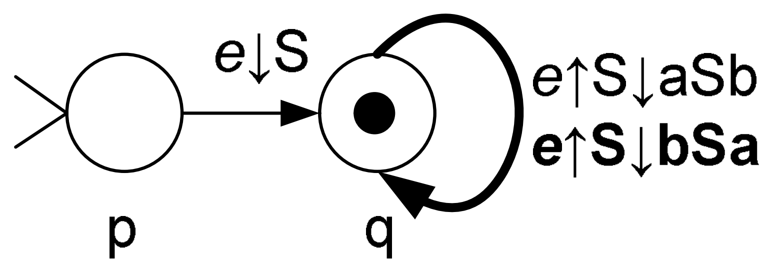
CFG → PDA: Example 1

- Step b1: introduce a new arc for S → aSb
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {**S → aSb** | bSa | *e*}



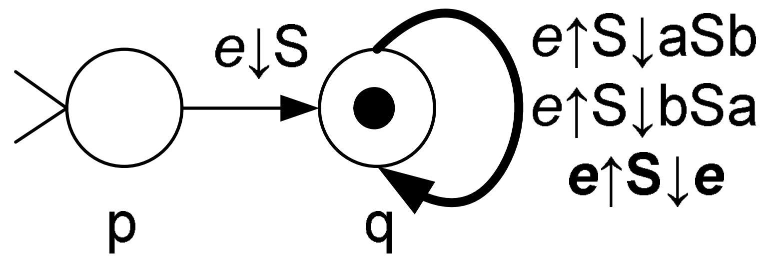
CFG → PDA: Example 1

- Step b2: introduce a new arc for S → bSa
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {**S** → aSb | **bSa** | *e*}



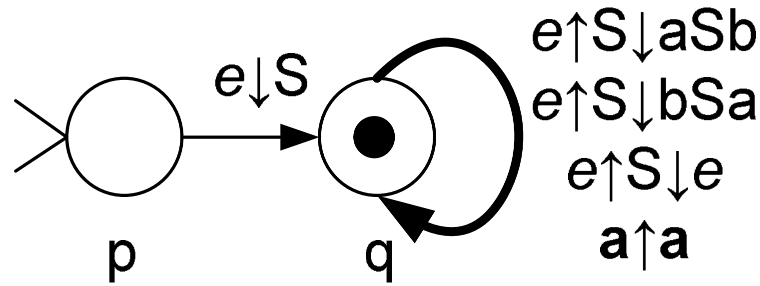
CFG → PDA: Example 1

- Step b3: introduce a new arc for S → *e*
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {**S** → aSb | bSa | ***e***}



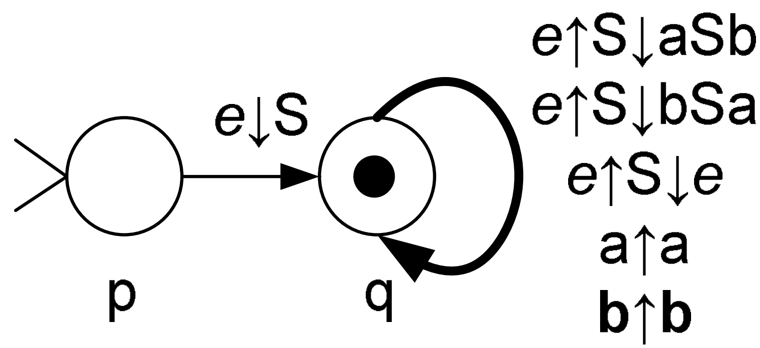
CFG → PDA: Example 1

- Step c1: introduce a new arc for a ∈ Σ
  - V = {a, b, S}
  - Σ = {**a**, b}
  - S = S
  - R = {S → aSb | bSa | *e*}



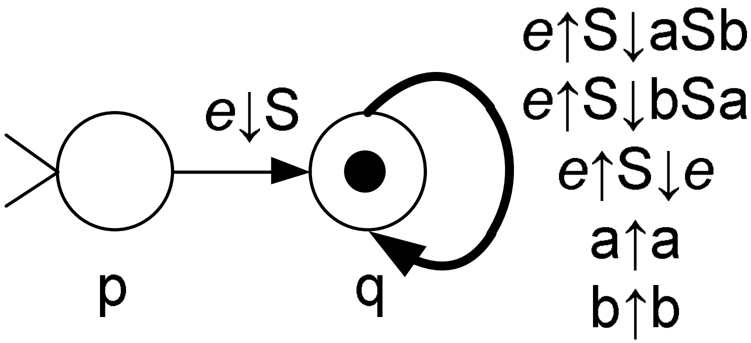
CFG → PDA: Example 1

- Step c2: introduce a new arc for b $\in \Sigma$
  - V = {a, b, S}
  - Σ = {a, **b**}
  - S = S
  - R = {S → aSb | bSa | *e*}



CFG → PDA: Example 1

- Example: abab
  - S => aSb => abSab => abab
  - (p, abab, *e*) |- (q, abab, S) |- (q, abab, aSb) |- (q, bab, Sb) |- (q, bab, bSab) |- (q, ab, Sab) |- (q, ab, ab) |- (q, b, b) |- (q, *e*, *e*)



CFG → PDA: Example 2

- Construct PDA M such that L(M) = L(G)! Use the set notations!
  - $V = \{a, b, S\}$
  - $\Sigma = \{a, b\}$
  - $S = S$
  - $R = \{S \rightarrow aSa \mid bSb \mid e\}$

CFG $\rightarrow$ PDA: Example 2

- Step a1: introduce an initial state
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S $\rightarrow$ aSa | bSb | *e*}
  - M = ({**p**, q}, Σ, V, Δ, **p**, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {}

CFG → PDA: Example 2

- Step a2: introduce a final state
  - $V = \{a, b, S\}$
  - $\Sigma = \{a, b\}$
  - $S = S$
  - $R = \{S \rightarrow aSa \mid bSb \mid e\}$
  - $M = (\{p, \mathbf{q}\}, \Sigma, V, \Delta, p, \mathbf{\{q\}})$
    - $\Sigma = \{a, b\}$
    - $V = \{S, a, b\}$
    - $\Delta = \{\}$

CFG $\rightarrow$ PDA: Example 2

- Step a3: initially push S into the stack
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S $\rightarrow$ aSa | bSb | *e*}
  - M = ({p, q}, Σ, V, Δ, p, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {**((p, *e*, *e*), (q, S))**}

CFG → PDA: Example 2

- Step b1: introduce a new arc for S → aSa
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → **aSa** | bSb | *e*}
  - M = ({p, q}, Σ, V, Δ, p, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {((p, *e*, *e*), (q, S)), **((q, *e*, S), (q, aSa))**}

CFG → PDA: Example 2

- Step b2: introduce a new arc for S → bSb
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aSa | **bSb** | *e*}

  - M = ({p, q}, Σ, V, Δ, p, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {((p, *e*, *e*), (q, S)), ((q, *e*, S), (q, aSa)), **((q, *e*, S), (q, bSb))**}

CFG → PDA: Example 2

- Step b3: introduce a new arc for S → *e*
  - V = {a, b, S}
  - Σ = {a, b}
  - S = S
  - R = {S → aSa | bSb | ***e***}

  - M = ({p, q}, Σ, V, Δ, p, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {((p, *e*, *e*), (q, S)), ((q, *e*, S), (q, aSa)), ((q, *e*, S), (q, bSb)), **((q, *e*, S), (q, *e*))}**

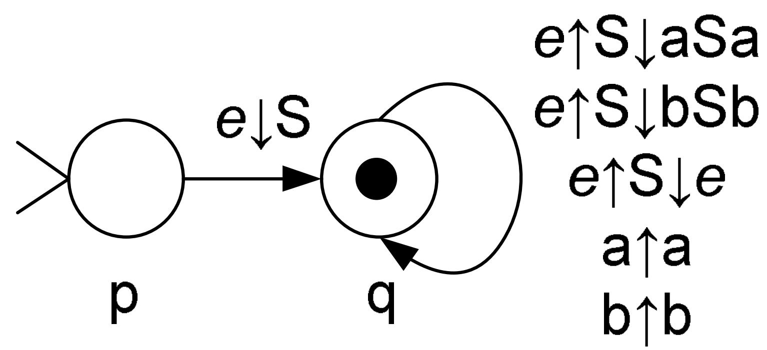
CFG → PDA: Example 2

- Step c1: introduce a new arc for a $\in \Sigma$
  - $V = \{a, b, S\}$
  - $\Sigma = \{\mathbf{a}, b\}$
  - $S = S$
  - $R = \{S \rightarrow aSa \mid bSb \mid e\}$

  - $M = (\{p, q\}, \Sigma, V, \Delta, p, \{q\})$
    - $\Sigma = \{a, b\}$
    - $V = \{S, a, b\}$
    - $\Delta = \{((p, e, e), (q, S)), ((q, e, S), (q, aSa)), ((q, e, S), (q, bSb)), ((q, e, S), (q, e)), \mathbf{((q, a, a), (q, \mathit{e}))}\}$

CFG → PDA: Example 2

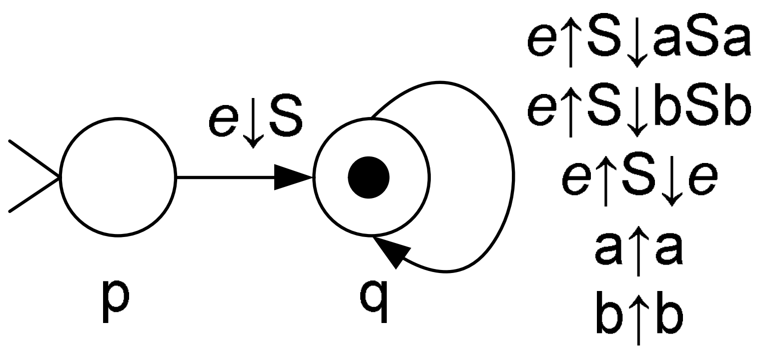
- Step c2: introduce a new arc for b $\in \Sigma$
  - V = {a, b, S}
  - Σ = {a, **b**}
  - S = S
  - R = {S → aSa | bSb | *e*}



  - M = ({p, q}, Σ, V, Δ, p, {q})
    - Σ = {a, b}
    - V = {S, a, b}
    - Δ = {((p, *e*, *e*), (q, S)), ((q, *e*, S), (q, aSa)), ((q, *e*, S), (q, bSb)), ((q, *e*, S), (q, *e*)), ((q, a, a), (q, *e*)), **((q, b, b), (q, *e*))**}

CFG → PDA: Example 2

- Example: abba
  - S => aSa => abSba => abba
  - (p, abba, *e*) |- (q, abba, S) |- (q, abba, aSa) |- (q, bba, Sa) |- (q, bba, bSba) |- (q, ba, Sba) |- (q, ba, ba) |- (q, a, a) |- (q, *e*, *e*)

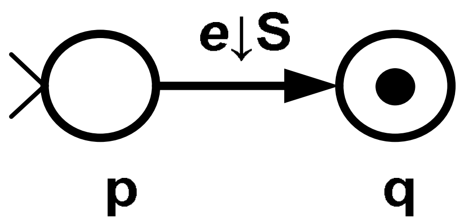


CFG $\rightarrow$ PDA: Example 3

- Construct PDA M such that L(M) = L(G)!
  - $V = \{a, b, c, d, S\}$
  - $\Sigma = \{a, b, c, d\}$
  - $S = S$
  - $R = \{S \rightarrow aSa \mid bSb \mid cSc \mid d\}$

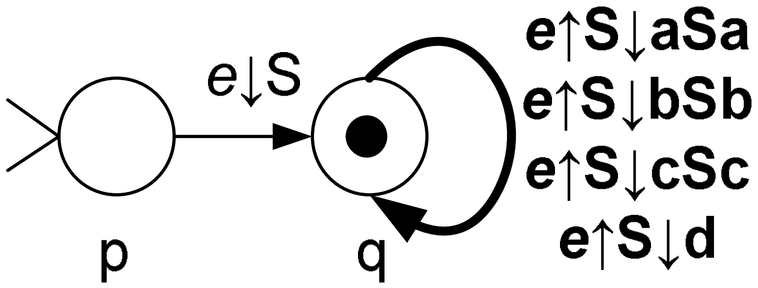
CFG → PDA: Example 3

- Step a: perform the initial steps
  - $V = \{a, b, c, d, S\}$
  - $\Sigma = \{a, b, c, d\}$
  - $S = S$
  - $R = \{S \rightarrow aSa \mid bSb \mid cSc \mid d\}$



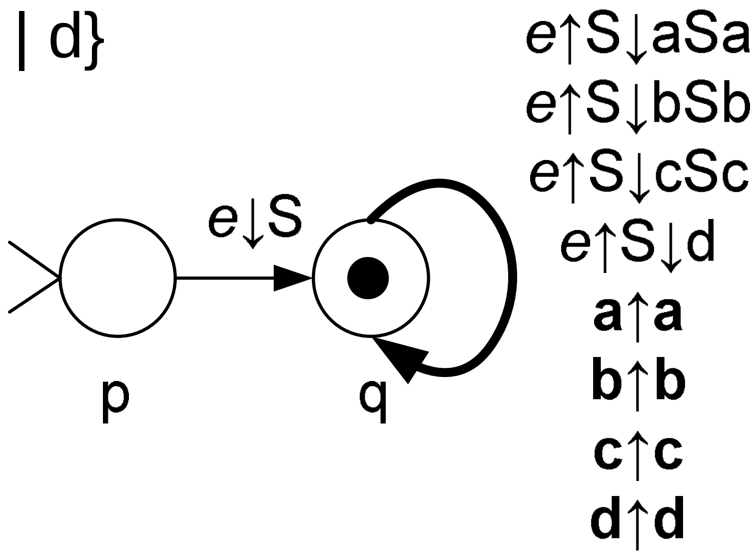
CFG → PDA: Example 3

- Step b: introduce new arcs for S → aSa | bSb | cSc | d
  - V = {a, b, c, d, S}
  - Σ = {a, b, c, d}
  - S = S
  - R = {S → aSa | bSb | cSc | d}



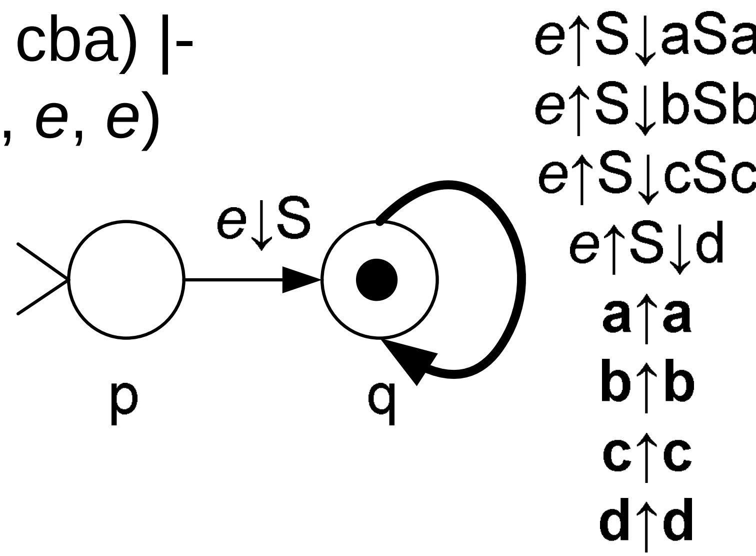
CFG → PDA: Example 3

- Step c: introduce a new arcs for elements of Σ
  - V = {a, b, c, d, S}
  - Σ = {a, b, c, d}
  - S = S
  - R = {S → aSa | bSb | cSc | d}



CFG → PDA: Example 3

- Example: abcdcba
  - S => aSa => abSba => abcScba => abcdcba
  - (p, abcdcba, *e*) |- (q, abcdcba, S) |-
    (q, abcdcba, aSa) |- (q, bcdcba, Sa) |-
    (q, bcdcba, bSba) |- (q, cdcba, Sba) |-
    (q, cdcba, cScba) |- (q, dcba, Scba) |-
    (q, dcba, dcba) |- (q, cba, cba) |-
    (q, ba, ba) |- (q, a, a) |- (q, *e*, *e*)



CFG → PDA: Example 4

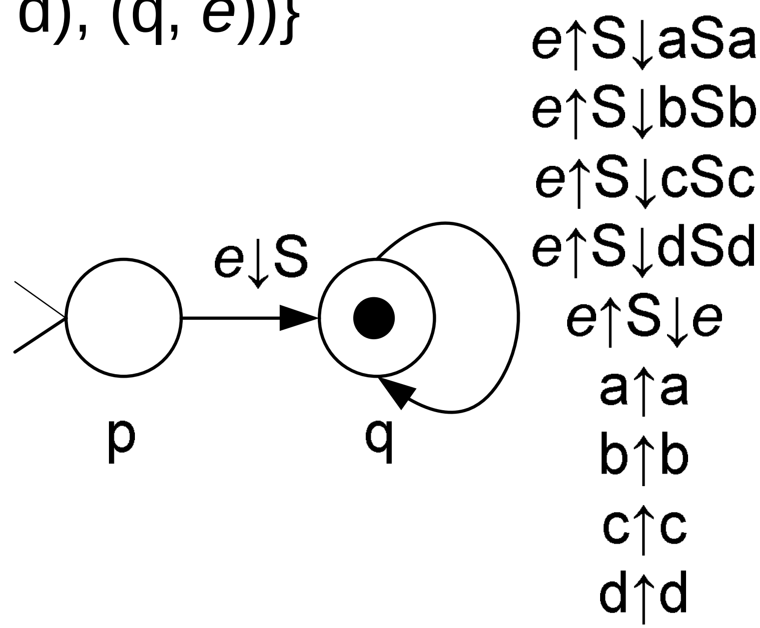
- Construct PDA M such that L(M) = L(G)! Use the set notations!
  - V = {a, b, c, d, S}
  - Σ = {a, b, c, d}
  - S = S
  - R = {S → aSa | bSb | cSc | dSd | *e*}

CFG $\rightarrow$ PDA: Example 4

- R = {$S \rightarrow aSa \mid bSb \mid cSc \mid dSd \mid e$}
- M = ({p, q}, Σ, V, Δ, p, {q})
  - Σ = {a, b, c, d, S}
  - V = { S, a, b, c, d}
  - Δ = {((p, *e*, *e*), (q, S)), ((q, *e*, S), (q, aSa)), ((q, *e*, S), (q, bSb)), ((q, *e*, S), (q, cSc)), ((q, *e*, S), (q, dSd)), ((q, *e*, S), (q, *e*)), ((q, a, a), (q, *e*)), ((q, b, b), (q, *e*)), ((q, c, c), (q, *e*)), ((q, d, d), (q, *e*))}

CFG → PDA: Example 4

- Give the sate diagram for M!
  - Δ = {((p, *e*, *e*), (q, S)), ((q, *e*, S), (q, aSa)), ((q, *e*, S), (q, bSb)), ((q, *e*, S), (q, cSc)), ((q, *e*, S), (q, dSd)), ((q, *e*, S), (q, *e*)), ((q, a, a), (q, *e*)), ((q, b, b), (q, *e*)), ((q, c, c), (q, *e*)), ((q, d, d), (q, *e*))}



CFG → PDA: Example 4

- Example: abcddcba
  - S => aSa => abSba => abcScba => abcdSdcba => abcddcba
  - (p, abcddcba, *e*) |- (q, abcddcba, S) |-
    (q, abcddcba, aSa) |- (q, bcddcba, Sa) |-
    (q, bcddcba, bSba) |- (q, cddcba, Sba) |-
    (q, cddcba, cScba) |- (q, ddcba, Scba) |-
    (q, ddcba, dSdcba) |- (q, dcba, Sdcba) |-
    (q, dcba, dcba) |- (q, cba, cba) |-
    (q, ba, ba) |- (q, a, a) |- (q, *e*, *e*)

Element of the Theory of the Computation

Lecture 10

Example 1: The T_{NEG} machine

- Create TM M which negates a binary number!
  - $\Sigma = \{\rhd, \sqcup, 0, 1\}$
  - e.g.: $\rhd\underline{\sqcup}1001 \rightarrow \rhd\sqcup0110\underline{\sqcup}$

Example 1: The T_{NEG} machine

- States:
  - q_0: start
  - q_1: negate
  - q_2: move forward
  - h_1: halt

Example 1: The T_{NEG} machine

- Transitions:
  - unused transitions are not displayed

q	σ	$\delta(q, \sigma)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_1	0	$(q_2, 1)$
q_1	1	$(q_2, 0)$
q_2	0 or 1	$(q_1, \rightarrow)$
q_1	$\sqcup$	$(h_1, \sqcup)$

States:

- q_0: start
- q_1: negate
- q_2: move forward
- h_1: halt

Example 1: The T_{NEG} machine

- Computation for $\rhd\underline{\sqcup}0101\sqcup$ without the yield in one step signs:
  - $(q_0, \rhd\underline{\sqcup}0101\sqcup)$
  - $(q_1, \rhd\sqcup\underline{0}101\sqcup)$
  - $(q_2, \rhd\sqcup\underline{1}101\sqcup)$
  - $(q_1, \rhd\sqcup1\underline{1}01\sqcup)$
  - $(q_2, \rhd\sqcup1\underline{0}01\sqcup)$
  - $(q_1, \rhd\sqcup10\underline{0}1\sqcup)$
  - $(q_2, \rhd\sqcup10\underline{1}1\sqcup)$
  - $(q_1, \rhd\sqcup101\underline{1}\sqcup)$
  - $(q_2, \rhd\sqcup101\underline{0}\sqcup)$
  - $(q_1, \rhd\sqcup1010\underline{\sqcup})$
  - $(h_1, \rhd\sqcup1010\underline{\sqcup})$

q	σ	$\delta(q, \sigma)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_1	0	$(q_2, 1)$
q_1	1	$(q_2, 0)$
q_2	0 or 1	$(q_1, \rightarrow)$
q_1	$\sqcup$	$(h_1, \sqcup)$

Example 2: The $L_{\sqcup}$ machine

- Create TM M, which scans for the first ⊔ to the left!
  - $\sum = \{\rhd, \sqcup, a, b, c\}$
  - e.g.: $\rhd\sqcup aacbb\underline{\sqcup} \rightarrow \rhd\underline{\sqcup}aacbb$
  - this TM never stops if there is no ⊔ to the left

Example 2: The $L_{\sqcup}$ machine

- States:
  - q_0: start
  - q_1: read
  - h_1: halt

Example 2: The $L_\sqcup$ machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	$\sqcup$	$(q_1, \leftarrow)$
q_1	a	$(q_1, \leftarrow)$
q_1	b	$(q_1, \leftarrow)$
q_1	c	$(q_1, \leftarrow)$
q_1	$\sqcup$	$(h_1, \sqcup)$

States:

- q_0: start
- q_1: read
- h_1: halt

Example 2: The $L_\sqcup$ machine

- Computation for $\rhd\sqcup abc\underline{\sqcup}$:
  - $(q_0, \rhd\sqcup abc\underline{\sqcup})$
  - $(q_1, \rhd\sqcup ab\underline{c}\sqcup)$
  - $(q_1, \rhd\sqcup a\underline{b}c\sqcup)$
  - $(q_1, \rhd\sqcup \underline{a}bc\sqcup)$
  - $(q_1, \rhd\underline{\sqcup}abc\sqcup)$
  - $(h_1, \rhd\underline{\sqcup}abc\sqcup)$

q	σ	δ(q, σ)
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_1	x	$(q_1, \rightarrow)$
q_1	y	$(q_1, \rightarrow)$
q_1	$\sqcup$	$(q_1, \rightarrow)$
q_1	z	(h_1, z)

Example 3: The R_z machine

- Create TM M, which scans for the first z to the right!
  - $\sum = \{\rhd, \sqcup, x, y, z\}$
  - e.g.: $\rhd\underline{\sqcup}xy\sqcup x\sqcup yzxx \rightarrow \rhd\sqcup xy\sqcup x\sqcup y\underline{z}xx$
  - this TM never stops if there is no z to the right

Example 3: The R_z machine

- States:
  - q_0: start
  - q_1: read
  - h_1: halt

Example 3: The R_z machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_1	x	$(q_1, \rightarrow)$
q_1	y	$(q_1, \rightarrow)$
q_1	$\sqcup$	$(q_1, \rightarrow)$
q_1	z	(h_1, z)

States:
- q_0: start
- q_1: read
- h_1: halt

Example 3: The R_z machine

- Computation for $\rhd\underline{\sqcup}xyzx$:
  - $(q_0, \rhd\underline{\sqcup}xyzx)$
  - $(q_1, \rhd\sqcup\underline{x}yzx)$
  - $(q_1, \rhd\sqcup x\underline{y}zx)$
  - $(q_1, \rhd\sqcup xy\underline{z}x)$
  - $(h_1, \rhd\sqcup xy\underline{z}x)$

q	σ	$\delta(q, \sigma)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_1	x	$(q_1, \rightarrow)$
q_1	y	$(q_1, \rightarrow)$
q_1	$\sqcup$	$(q_1, \rightarrow)$
q_1	z	(h_1, z)

Example 4: The c machine

- Create TM M, which writes c to the actual position!
  - $\sum = \{\triangleright, \sqcup, a, b, c\}$
  - e.g.: $\triangleright\sqcup aa\underline{a}bb \rightarrow \triangleright\sqcup aa\underline{c}bb$

Example 4: The c machine

- States:
  - q_0: start
  - h_1: halt

Example 4: The c machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	a	(h_1, c)
q_0	b	(h_1, c)
q_0	c	(h_1, c)

States:

- q_0: start
- h_1: halt

Example 4: The c machine

- Computation for ⊳aa$\underline{b}$bcc:
  - (q_0, ⊳aa$\underline{b}$bcc)
  - (h_1, ⊳aa$\underline{c}$bcc)

q	σ	δ(q, σ)
q_0	a	(h_1, c)
q_0	b	(h_1, c)
q_0	c	(h_1, c)

Example 5: Create the RR machine

- Create TM M, which steps two positions to the right!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - e.g.: $\rhd\sqcup aa\underline{a}bb \rightarrow \rhd\sqcup aaab\underline{b}$

Example 5: Create the RR machine

- States:
  - q_0: two more right
  - q_1: one more right
  - h_1: halt

Example 5: Create the RR machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	$\rhd$	$(q_1, \rightarrow)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_0	a	$(q_1, \rightarrow)$
q_0	b	$(q_1, \rightarrow)$
q_1	a	$(h_1, \rightarrow)$
q_1	b	$(h_1, \rightarrow)$
q_1	$\sqcup$	$(h_1, \rightarrow)$

States:

- q_0: two more right
- q_1: one more right
- h_1: halt

Example 5: Create the RR machine

- Computation for $\underline{\rhd}\sqcup aba$:
  - $(q_0, \underline{\rhd}\sqcup aba)$
  - $(q_1, \rhd\underline{\sqcup}aba)$
  - $(h_1, \rhd\sqcup\underline{a}ba)$
- Computation $\rhd\sqcup a\underline{b}a$:
  - $(q_0, \rhd\sqcup a\underline{b}a)$
  - $(q_1, \rhd\sqcup ab\underline{a})$
  - $(h_1, \rhd\sqcup aba\underline{\sqcup})$

q	σ	$\delta(q, \sigma)$
q_0	$\rhd$	$(q_1, \rightarrow)$
q_0	$\sqcup$	$(q_1, \rightarrow)$
q_0	a	$(q_1, \rightarrow)$
q_0	b	$(q_1, \rightarrow)$
q_1	a	$(h_1, \rightarrow)$
q_1	b	$(h_1, \rightarrow)$
q_1	$\sqcup$	$(h_1, \rightarrow)$

Example 6: Create the La machine

- Create TM M, which steps to the left and writes a!
  - ∑ = {⊳, ⊔, a, b}
  - e.g.: ⊳⊔ab<u>a</u>bb → ⊳⊔a<u>b</u>abb → ⊳⊔a<u>a</u>abb
- What should we do on ⊳?
  - step to right and halt

Example 6: Create the La machine

- States:
  - q_0: go left
  - q_1: write 'a'
  - h_1: halt

Example 6: Create the La machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	$\rhd$	$(h_1, \rightarrow)$
q_0	$\sqcup$	$(q_1, \leftarrow)$
q_0	a	$(q_1, \leftarrow)$
q_0	b	$(q_1, \leftarrow)$
q_1	$\rhd$	$(h_1, \rightarrow)$
q_1	a	(h_1, a)
q_1	b	(h_1, a)
q_1	$\sqcup$	(h_1, a)

States:

- q_0: go left
- q_1: write 'a'
- h_1: halt

Example 6: Create the La machine

- Computation for $\rhd \sqcup ab\underline{b}$:
  - $(q_0, \rhd \sqcup ab\underline{b})$
  - $(q_1, \rhd \sqcup a\underline{b}b)$
  - $(h_1, \rhd \sqcup a\underline{a}b)$
- Computation for $\rhd \underline{\sqcup} abb$:
  - $(q_0, \rhd \underline{\sqcup} abb)$
  - $(q_1, \underline{\rhd} \sqcup abb)$
  - $(h_1, \rhd \underline{\sqcup} abb)$

q	σ	δ(q, σ)
q_0	$\rhd$	$(h_1, \rightarrow)$
q_0	$\sqcup$	$(q_1, \leftarrow)$
q_0	a	$(q_1, \leftarrow)$
q_0	b	$(q_1, \leftarrow)$
q_1	$\rhd$	$(h_1, \rightarrow)$
q_1	a	(h_1, a)
q_1	b	(h_1, a)
q_1	$\sqcup$	(h_1, a)

Example 7: Create the RbR machine

- Create TM M, which steps to the right, writes b, and steps to right again!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - e.g.: $\rhd\sqcup a\underline{b}aab \rightarrow \rhd\sqcup ab\underline{a}ab \rightarrow \rhd\sqcup abb\underline{a}b$

Example 7: Create the RbR machine

- States:
  - q_0: go right
  - q_1: write 'b'
  - q_2: go right
  - h_1: halt

Example 7: Create the RbR machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_0	x	$(q_1, \rightarrow)$
q_1	y	(q_2, b)
q_2	y	$(h_1, \rightarrow)$

$x \in \{a, b, \rhd, \sqcup\}$

$y \in \{a, b, \sqcup\}$

- States:
  - q_0: go right
  - q_1: write 'b'
  - q_2: go right
  - h_1: halt

Example 7: Create the RbR machine

- Computation for $\rhd\sqcup a\underline{b}a$:
  - $(q_0, \rhd\sqcup a\underline{b}a)$
  - $(q_1, \rhd\sqcup ab\underline{a})$
  - $(q_2, \rhd\sqcup ab\underline{b})$
  - $(h_1, \rhd\sqcup aab\underline{\sqcup})$

q	σ	$\delta(q, \sigma)$
q_0	x	$(q_1, \rightarrow)$
q_1	y	(q_2, b)
q_2	y	$(h_1, \rightarrow)$

Example 8: Sort machine

- Create a TM M, which arranges all 'b' in a word to the beginning of the word!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - e.g.: $\rhd\underline{\sqcup}abaab \rightarrow \rhd\sqcup bbaaa\underline{\sqcup}$

Example 8: Sort machine

- Logic of the operation:
  - find b to the right
  - move this b to the beginning part of the string by consecutively swapping b and the preceding a
  - find the next b
  - e.g.:
    - bbaaabaab, we are in the middle of the algorithm
    - bbaaabaab, b is found
    - bbaabaaab, first swap
    - ...
    - bbbaaaaab, last swap, b is at the final position
    - bbbaaaaab, start of the next cycle

Example 8: Sort machine

```
move to the right
while not blank
   if b is found
      move to the left
      while a is found
         swap a and b, head on b
         move to the left
      move to the right
      move to the right
   move to the right
```

- The transition function is not implemented exactly according to the algorithm

Example 8: Sort machine

- States:
  - q_i: initial state, we has to move to the right
  - q_0: go to the right until b or ⊔
  - q_1: b is found
  - q_2: a $\rightarrow$ b change is performed
  - q_3: right move is done
  - q_4: b $\rightarrow$ a change is performed
  - q_5: the character before b was not 'a'
  - h_1: halt

Example 8: Sort machine

- Transitions:

q	σ	$\delta(q, \sigma)$
q_i	$\sqcup$	$(q_0, \rightarrow)$
q_0	a	$(q_0, \rightarrow)$
q_0	$\sqcup$	$(h_1, \sqcup)$
q_0	b	$(q_1, \leftarrow)$
q_1	a	(q_2, b)
q_2	b	$(q_3, \rightarrow)$
q_3	b	(q_4, a)
q_4	a	$(q_0, \leftarrow)$
q_1	b	$(q_5, \rightarrow)$
q_1	$\sqcup$	$(q_5, \rightarrow)$
q_5	b	$(q_0, \rightarrow)$

States:

- q_i: initial state, we has to move to the right
- q_0: go to the right until b or $\sqcup$
- q_1: b is found
- q_2: a $\rightarrow$ b change is performed
- q_3: right move is done
- q_4: b $\rightarrow$ a change is performed
- q_5: the character before b was not 'a'
- h_1: halt

It is not easy to represent and understand δ

- if the algorithm was followed then more states would be required

Example 8: Sort machine

- Computation:
  - (q_i, $\rhd\underline{\sqcup}ababaa\sqcup$), search for b
  - (q_0, $\rhd\sqcup\underline{a}babaa\sqcup$)
  - (q_0, $\rhd\sqcup a\underline{b}abaa\sqcup$), b is found
  - (q_1, $\rhd\sqcup\underline{a}babaa\sqcup$), character before b is checked
  - (q_2, $\rhd\sqcup\underline{b}babaa\sqcup$), a $\rightarrow$ b
  - (q_3, $\rhd\sqcup b\underline{b}abaa\sqcup$)
  - (q_4, $\rhd\sqcup b\underline{a}abaa\sqcup$), b $\rightarrow$ a
  - (q_0, $\rhd\sqcup\underline{b}aabaa\sqcup$), b is found
  - (q_1, $\rhd\underline{\sqcup}baabaa\sqcup$), character before b is checked

Example 8: Sort machine

– (q_5, ⊳⊔$\underline{\text{b}}$aabaa⊔), swap was not needed
– (q_0, ⊳⊔b$\underline{\text{a}}$abaa⊔), first b is in position
…
– (q_0, ⊳⊔baa$\underline{\text{b}}$aa⊔)
– (q_1, ⊳⊔ba$\underline{\text{a}}$baa⊔)
– (q_2, ⊳⊔ba$\underline{\text{b}}$baa⊔)
– (q_3, ⊳⊔bab$\underline{\text{b}}$aa⊔)

Example 8: Sort machine

- $(q_4, \rhd\sqcup bab\underline{a}aa\sqcup)$
- $(q_0, \rhd\sqcup ba\underline{b}aaa\sqcup)$
- $(q_1, \rhd\sqcup b\underline{a}baaa\sqcup)$

  …
- $(q_0, \rhd\sqcup b\underline{b}aaaa\sqcup)$
- $(q_1, \rhd\sqcup \underline{b}baaaa\sqcup)$
- $(q_5, \rhd\sqcup b\underline{b}aaaa\sqcup)$
- $(q_0, \rhd\sqcup bb\underline{a}aaa\sqcup)$, second b is in position

  …
- $(q_0, \rhd\sqcup bbaaaa\underline{\sqcup})$, searching for b
- $(h_1, \rhd\sqcup bbaaaa\underline{\sqcup})$

Turing Machines with 2 tape

- A Turing Machine can have 2 or more tapes
  - such a machine can be simulated with a regular TM
- A TM with two tapes has transition function like $\delta(q, a_1, a_2) = (p, b_1, b_2)$
  - if TM M is in state q and reads a_1 from the first tape, a_2 from the second tape, then it goes to state p and writes b_1 to the first tape, b_2 to the second tape

Example 9: Copy machine

- Create TM M with two tapes, which copies a word from the first tape to the second tape!
  - $\sum = \{\rhd, \sqcup, a, b, c\}$
  - e.g.: T_1: $\rhd\underline{\sqcup}abaab$, T_2: $\rhd\underline{\sqcup}$ $\rightarrow$ T_1: $\rhd\underline{\sqcup}abaab$, T_2: $\rhd\sqcup abaab\underline{\sqcup}$

Example 9: Copy machine

- States:
  - q_i: we have to move to the right
  - q_0: we moved to the right
  - q_1: character is copied
  - h_1: halt

Example 9: Copy machine

- Transitions:

q	σ_1	σ_2	$\delta(q, \sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	$\rhd$	$\rhd$	$(q_0, \rightarrow, \rightarrow)$
q_0	a	$\sqcup$	(q_1, a, a)
q_0	b	$\sqcup$	(q_1, b, b)
q_0	c	$\sqcup$	(q_1, c, c)
q_1	a	a	$(q_0, \rightarrow, \rightarrow)$
q_1	b	b	$(q_0, \rightarrow, \rightarrow)$
q_1	c	c	$(q_0, \rightarrow, \rightarrow)$
q_0	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$

States:

- q_i: we have to move to the right
- q_0: we moved to the right
- q_1: character is copied
- h_1: halt

Example 9: Copy machine

- Computation:
  - $(q_i, \triangleright\underline{\sqcup}abc\sqcup, \triangleright\underline{\sqcup}\sqcup\sqcup\sqcup\sqcup)$
  - $(q_0, \triangleright\sqcup\underline{a}bc\sqcup, \triangleright\sqcup\underline{\sqcup}\sqcup\sqcup\sqcup)$
  - $(q_1, \triangleright\sqcup\underline{a}bc\sqcup, \triangleright\sqcup\underline{a}\sqcup\sqcup\sqcup)$
  - $(q_0, \triangleright\sqcup a\underline{b}c\sqcup, \triangleright\sqcup a\underline{\sqcup}\sqcup\sqcup)$

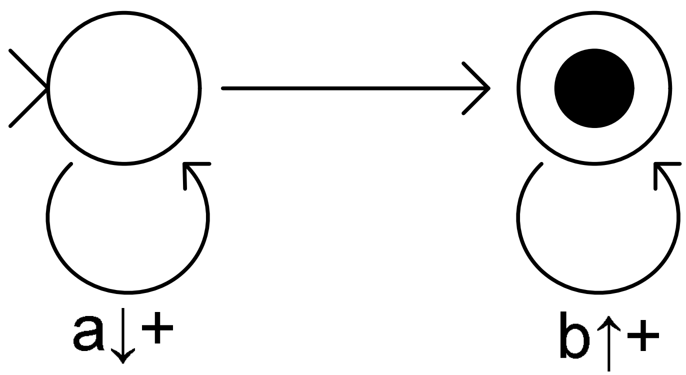
    …
  - $(q_1, \triangleright\sqcup ab\underline{c}\sqcup, \triangleright\sqcup ab\underline{c}\sqcup)$
  - $(q_0, \triangleright\sqcup abc\underline{\sqcup}, \triangleright\sqcup abc\underline{\sqcup})$
  - $(h_1, \triangleright\sqcup abc\underline{\sqcup}, \triangleright\sqcup abc\underline{\sqcup})$

q	σ_1	σ_2	$\delta(q, \sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	$\triangleright$	$\triangleright$	$(q_0, \rightarrow, \rightarrow)$
q_0	a	$\sqcup$	(q_1, a, a)
q_0	b	$\sqcup$	(q_1, b, b)
q_0	c	$\sqcup$	(q_1, c, c)
q_1	a	a	$(q_0, \rightarrow, \rightarrow)$
q_1	b	b	$(q_0, \rightarrow, \rightarrow)$
q_1	c	c	$(q_0, \rightarrow, \rightarrow)$
q_0	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$

Example 10: Simulate PDA with TM

- Create such a TM that simulates PDA M!
  - PDA M accepts $L(M) = \{w \in a^n b^n\}$
  - e.g.: $\rhd\underline{\sqcup}aabb \rightarrow h_1$ (accepting halting state)

- PDA:
  - '+' symbols indicates the number of 'a'



Example 10: Simulate PDA with TM

- The TM requires two tapes
  - tape 1: input tape, we move only to the right on this tape
  - tape 2: stack, '+' symbols indicates the number of 'a'

Example 10: Simulate PDA with TM

- States:
  - q_i: we have to move to the right
  - q_0: input has to be read, '+' is pushed
  - q_1: 'a' is read
  - q_2: pop is finished
  - q_3: 'b' is read, '+' is deleted
  - h_1: string accepted
  - h_2: $|b|>|a|$
  - h_3: $|a|>|b|$
  - h_4: 'a' is read after 'b'

A state can be reached after different actions, different actions can be performed after a state, thus a state can have different names

Example 10: Simulate PDA with TM

- Transitions:

q	σ_1	σ_2	$\delta(\sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	a	$\sqcup$	$(q_1, \rightarrow, \sqcup)$
q_0	a	+	$(q_1, \rightarrow, \rightarrow)$
q_1	a	$\sqcup$	$(q_0, a, +)$
q_1	b	$\sqcup$	$(q_0, b, +)$
q_2	b	+	$(q_3, \rightarrow, \sqcup)$
q_3	b	$\sqcup$	$(q_2, b, \leftarrow)$
q_3	$\sqcup$	$\sqcup$	$(q_2, \sqcup, \leftarrow)$

States:

- q_i: we have to move to the right
- q_0: input has to be read, '+' is pushed
- q_1: 'a' is read
- q_2: pop is finished
- q_3: 'b' is read, '+' is deleted
- h_1: string accepted
- h_2: $|b|>|a|$
- h_3: $|a|>|b|$
- h_4: 'a' is read after 'b'

Example 10: Simulate PDA with TM

- Transitions:

q_2	⊔	⊔	(h_1, ⊔, ⊔)
q_0	⊔	⊔	(h_1, ⊔, ⊔)
q_0	b	⊔	(h_2, b, ⊔)
q_2	b	⊔	(h_2, b, ⊔)
q_2	⊔	+	(h_3, ⊔, +)
q_2	a	+	(h_4, a, +)

States:

- q_i: we have to move to the right
- q_0: input has to be read, '+' is pushed
- q_1: 'a' is read
- q_2: pop is finished
- q_3: 'b' is read, '+' is deleted
- h_1: string accepted
- h_2: |b|>|a|
- h_3: |a|>|b|
- h_4: 'a' is read after 'b'

Example 10: Simulate PDA with TM

- Computation:
  - $(q_i, \triangleright\underline{\sqcup}aabb\sqcup, \triangleright\underline{\sqcup})$
  - $(q_0, \triangleright\sqcup\underline{a}abb\sqcup, \triangleright\sqcup\underline{\sqcup})$
  - $(q_1, \triangleright\sqcup a\underline{a}bb\sqcup, \triangleright\sqcup\underline{\sqcup})$
  - $(q_0, \triangleright\sqcup a\underline{a}bb\sqcup, \triangleright\sqcup\underline{+})$, push 0
  - $(q_1, \triangleright\sqcup aa\underline{b}b\sqcup, \triangleright\sqcup+\underline{\sqcup})$
  - $(q_0, \triangleright\sqcup aa\underline{b}b\sqcup, \triangleright\sqcup+\underline{+})$, push 2

q	σ_1	σ_2	$\delta(\sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	a	$\sqcup$	$(q_1, \rightarrow, \sqcup)$
q_0	a	+	$(q_1, \rightarrow, \rightarrow)$
q_1	a	$\sqcup$	$(q_0, a, +)$
q_1	b	$\sqcup$	$(q_0, b, +)$
q_2	b	+	$(q_3, \rightarrow, \sqcup)$
q_3	b	$\sqcup$	$(q_2, b, \leftarrow)$
q_3	$\sqcup$	$\sqcup$	$(q_2, \sqcup, \leftarrow)$
q_0	b	+	$(q_2, b, +)$

q_2	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$
q_0	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$
q_0	b	$\sqcup$	$(h_2, b, \sqcup)$
q_2	b	$\sqcup$	$(h_2, b, \sqcup)$
q_2	$\sqcup$	+	$(h_3, \sqcup, +)$
q_2	a	+	$(h_4, a, +)$
q_2	a	$\sqcup$	$(h_4, a, \sqcup)$

Example 10: Simulate PDA with TM

- Computation:
  - $(q_2, \rhd\sqcup aa\underline{b}b\sqcup, \rhd\sqcup+\underline{+}\sqcup)$
  - $(q_3, \rhd\sqcup aab\underline{b}\sqcup, \rhd\sqcup+\underline{\sqcup}\sqcup)$
  - $(q_2, \rhd\sqcup aab\underline{b}\sqcup, \rhd\sqcup\underline{+}\sqcup\sqcup)$, pop 1
  - $(q_3, \rhd\sqcup aabb\underline{\sqcup}, \rhd\sqcup\underline{\sqcup}\sqcup\sqcup)$
  - $(q_2, \rhd\sqcup aabb\underline{\sqcup}, \rhd\underline{\sqcup}\sqcup\sqcup\sqcup)$, pop 2
  - $(h_1, \rhd\sqcup aabb\underline{\sqcup}, \rhd\underline{\sqcup}\sqcup\sqcup\sqcup)$

q	σ_1	σ_2	$\delta(\sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	a	$\sqcup$	$(q_1, \rightarrow, \sqcup)$
q_0	a	+	$(q_1, \rightarrow, \rightarrow)$
q_1	a	$\sqcup$	$(q_0, a, +)$
q_1	b	$\sqcup$	$(q_0, b, +)$
q_2	b	+	$(q_3, \rightarrow, \sqcup)$
q_3	b	$\sqcup$	$(q_2, b, \leftarrow)$
q_3	$\sqcup$	$\sqcup$	$(q_2, \sqcup, \leftarrow)$
q_0	b	+	$(q_2, b, +)$

q_2	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$
q_0	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$
q_0	b	$\sqcup$	$(h_2, b, \sqcup)$
q_2	b	$\sqcup$	$(h_2, b, \sqcup)$
q_2	$\sqcup$	+	$(h_3, \sqcup, +)$
q_2	a	+	$(h_4, a, +)$
q_2	a	$\sqcup$	$(h_4, a, \sqcup)$

Example 11: The T_{ADD} machine

- Create TM M with two tapes, which adds 2 binary numbers!
  - $\sum = \{\rhd, \sqcup, 0, 1\}$
  - the binary numbers are written in reverse
  - the numbers should have the same length
    - use extra zeros if needed
  - the result should be on the first tape
  - e.g.: T1: 0111, T2: 1010 $\rightarrow$ T1: 11001
    - $14_{10} + 5_{10} = 19_{10} \rightarrow 1110_2 + 0101_2 = 10011_2$

Example 11: The T_{ADD} machine

- States:
  - q_i: start
  - q_0: add, no carry
  - q_1: move right, no carry
  - q_2: add, carry
  - q_3: move right, carry
  - h_1: halt

Example 11: The T_{ADD} machine

- Transitions:

q	σ_1	σ_2	$\delta(q, \sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	0	0	$(q_1, 0, 0)$
q_0	0	1	$(q_1, 1, 1)$
q_0	1	0	$(q_1, 1, 0)$
q_0	1	1	$(q_3, 0, 1)$
q_2	0	0	$(q_1, 1, 0)$
q_2	0	1	$(q_3, 0, 1)$
q_2	1	0	$(q_3, 0, 0)$
q_2	1	1	$(q_3, 1, 1)$

States:

- q_i: start
- q_0: add, no carry
- q_1: move right, no carry
- q_2: add, carry
- q_3: move right, carry
- h_1: halt

Example 11: The T_{ADD} machine

- Transitions:

q_1	0 or 1	0 or 1	$(q_0, \rightarrow, \rightarrow)$
q_3	0 or 1	0 or 1	$(q_2, \rightarrow, \rightarrow)$
q_2	⊔	⊔	$(h_1, 1, ⊔)$
q_0	⊔	⊔	$(h_1, ⊔, ⊔)$

States:

- q_i: start
- q_0: add, no carry
- q_1: move right, no carry
- q_2: add, carry
- q_3: move right, carry
- h_1: halt

Example 11: The T_{ADD} machine

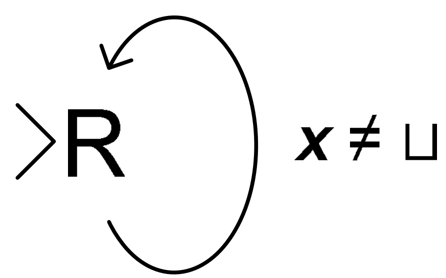
q	σ_1	σ_2	$\delta(q, \sigma_1, \sigma_2)$
q_i	$\sqcup$	$\sqcup$	$(q_0, \rightarrow, \rightarrow)$
q_0	0	0	$(q_1, 0, 0)$
q_0	0	1	$(q_1, 1, 1)$
q_0	1	0	$(q_1, 1, 0)$
q_0	1	1	$(q_3, 0, 1)$
q_2	0	0	$(q_1, 1, 0)$
q_2	0	1	$(q_3, 0, 1)$
q_2	1	0	$(q_3, 0, 0)$
q_2	1	1	$(q_3, 1, 1)$

- $(q_i, \rhd\underline{\sqcup}0111, \rhd\underline{\sqcup}1010)$
- $(q_0, \rhd\sqcup\underline{0}111, \rhd\sqcup\underline{1}010)$ // 0 + 1
- $(q_1, \rhd\sqcup\underline{1}111, \rhd\sqcup\underline{1}010)$
- $(q_0, \rhd\sqcup 1\underline{1}11, \rhd\sqcup 1\underline{0}10)$ // 1 + 0
- $(q_1, \rhd\sqcup 1\underline{1}11, \rhd\sqcup 1\underline{0}10)$
- $(q_0, \rhd\sqcup 11\underline{1}1, \rhd\sqcup 10\underline{1}0)$ // 1 + 1
- $(q_3, \rhd\sqcup 11\underline{0}1, \rhd\sqcup 10\underline{1}0)$
- $(q_2, \rhd\sqcup 110\underline{1}, \rhd\sqcup 101\underline{0})$ // 1 + 0 + carry
- $(q_3, \rhd\sqcup 110\underline{0}, \rhd\sqcup 101\underline{0})$
- $(q_2, \rhd\sqcup 1100\underline{\sqcup}, \rhd\sqcup 1010\underline{\sqcup})$ // carry
- $(h_1, \rhd\sqcup 1100\underline{1}, \rhd\sqcup 1010\underline{\sqcup})$ // halt

q_1	0 or 1	0 or 1	$(q_0, \rightarrow, \rightarrow)$
q_3	0 or 1	0 or 1	$(q_2, \rightarrow, \rightarrow)$
q_2	$\sqcup$	$\sqcup$	$(h_1, 1, \sqcup)$
q_0	$\sqcup$	$\sqcup$	$(h_1, \sqcup, \sqcup)$

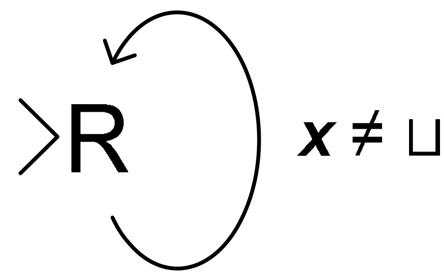
Example 12: The $R_{\sqcup}$ machine

- Create a machine schema, which scans for the first $\sqcup$ to the right!



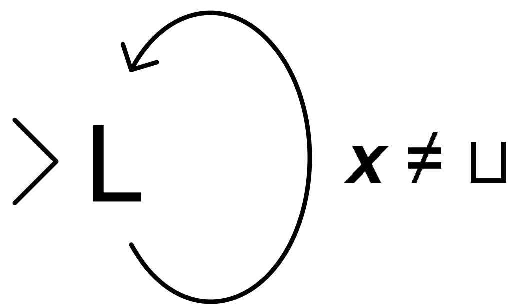
Example 12: The $R_{\sqcup}$ machine

- Computation for $\underline{\sqcup}abc$:
  - $\underline{\sqcup}abc$
  - $\sqcup\underline{a}bc$
  - $\sqcup a\underline{b}c$
  - $\sqcup ab\underline{c}$
  - $\sqcup abc\underline{\sqcup}$



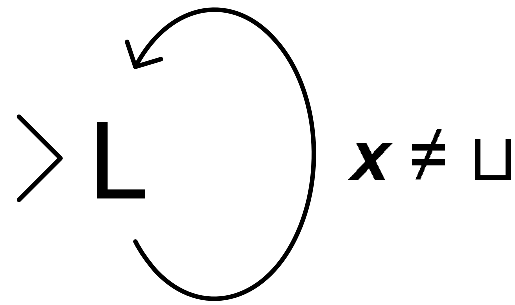
Example 13: The $L_{\sqcup}$ machine

- Create a machine schema, which scans for the first ⊔ to the left!
  – this TM never stops if there is no ⊔ to the left



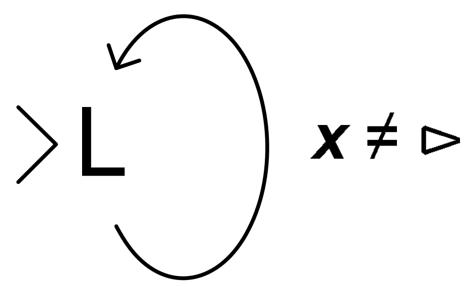
Example 13: The $L_{\sqcup}$ machine

- Computation for $\sqcup abc\underline{\sqcup}$:
  - $\sqcup abc\underline{\sqcup}$
  - $\sqcup ab\underline{c}$
  - $\sqcup a\underline{b}c$
  - $\sqcup \underline{a}bc$
  - $\underline{\sqcup}abc$
- Computation for $\rhd a\underline{b}c$:
  - $\rhd a\underline{b}c$
  - $\rhd \underline{a}bc$
  - $\underline{\rhd}abc$
  - $\rhd \underline{a}bc$
  - $\underline{\rhd}abc$ // loop again



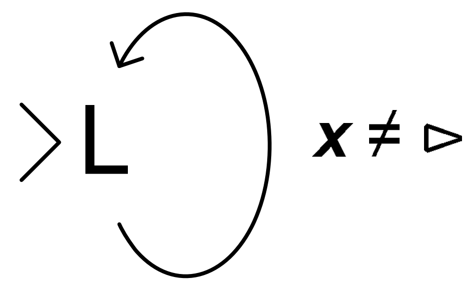
Example 14: The $L_{\triangleright}$ machine

- Create a machine, which goes to the beginning of the tape!
  - this machine never stops



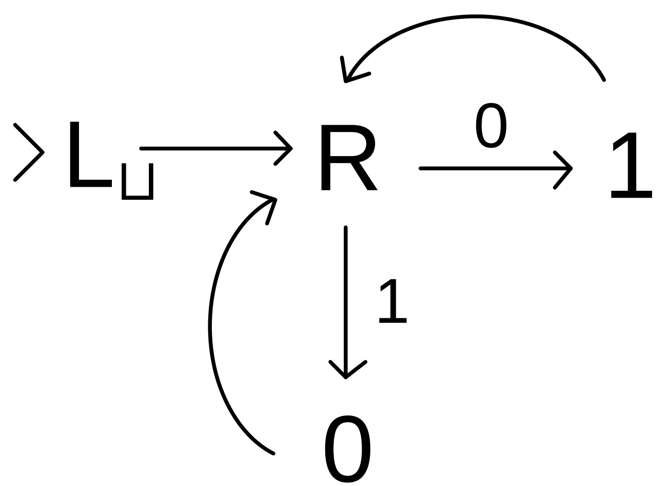
Example 14: The $L_{\rhd}$ machine

- Computation for $\rhd\sqcup 0\underline{1}2$:
  - $\rhd\sqcup 0\underline{1}2$
  - $\rhd\sqcup \underline{0}12$
  - $\rhd\underline{\sqcup}012$
  - $\underline{\rhd}\sqcup 012$
  - $\rhd\underline{\sqcup}012$
  - $\underline{\rhd}\sqcup 012$ // loop again



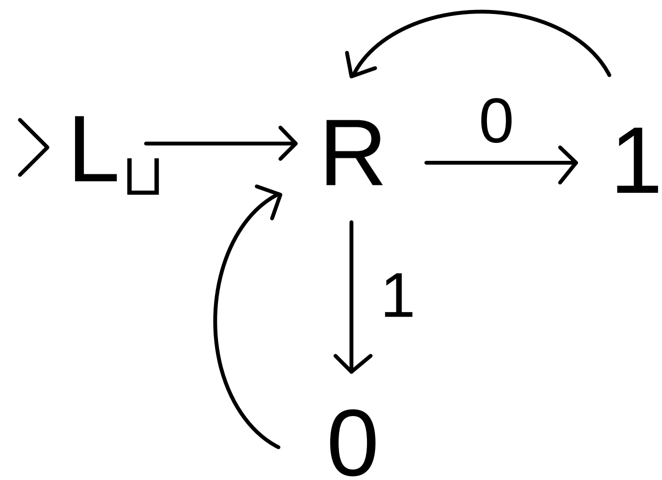
Example 15: The T_{NEG} machine

- Create a machine schema, which negates a binary number!
  - $\sum = \{\rhd, \sqcup, 0, 1\}$
  - e.g.: $\sqcup 0111\underline{\sqcup} \rightarrow \sqcup 1000\underline{\sqcup}$



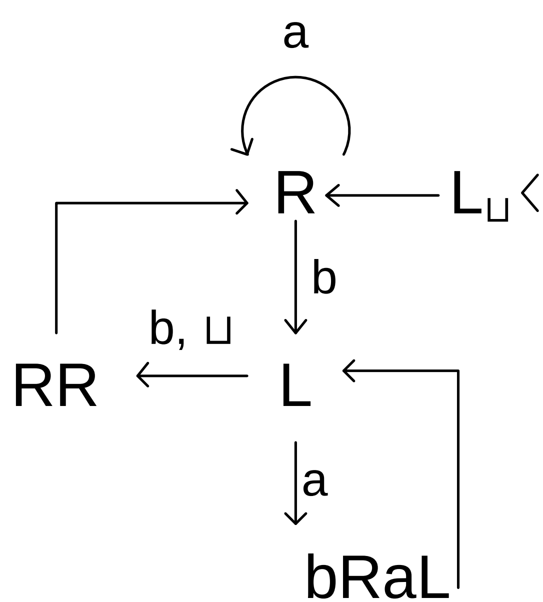
Example 15: The T_{NEG} machine

- Computation for $\sqcup 0101\underline{\sqcup}$:
  - $\sqcup 0101\underline{\sqcup}$
  - $\underline{\sqcup}0101$
  - $\sqcup\underline{0}101$
  - $\sqcup\underline{1}101$
  - $\sqcup 1\underline{1}01$
  - $\sqcup 1\underline{0}01$
  - $\sqcup 10\underline{0}1$
  - $\sqcup 10\underline{1}1$
  - $\sqcup 101\underline{1}$
  - $\sqcup 101\underline{0}$
  - $\sqcup 1010\underline{\sqcup}$



Example 16: Sort machine

- Create a machine schema, which arranges all 'b' in a word to the beginning of the word!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - e.g.: $\sqcup abaab\underline{\sqcup} \rightarrow \sqcup bbaaa\underline{\sqcup}$

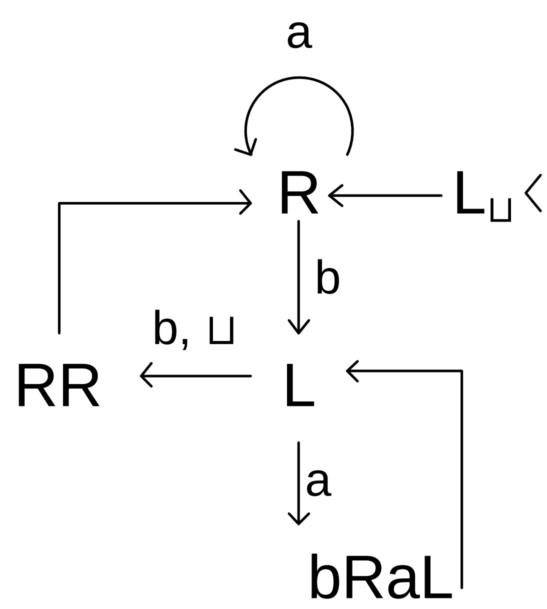


Example 16: Sort machine

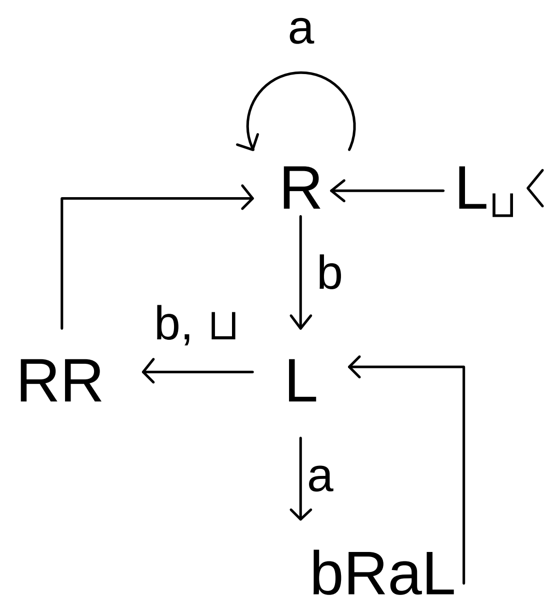
- Logic of the operation:
  - find b to the right
  - move this b to the beginning part of the string by consecutively swapping b and the preceding a
  - find the next b to the right
  - e.g.:
    - bbaaabaab, we are in the middle of the algorithm
    - bbaaabaab, b is found
    - bbaabaaab, first swap
    - ...
    - bbbaaaaab, last swap, b is at the final position
    - bbbaaaaab, start again, find b to the right

Example 16: Sort machine

```
move to the right
while not blank
   if b is found
      move to the left
      while a is found
         swap a and b, head on b
         move to the left
      move to the right
      move to the right
   move to the right
```



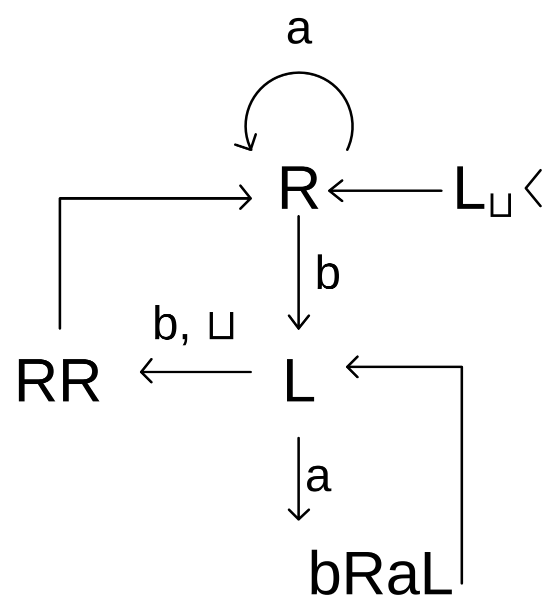
Example 16: Sort machine



- Computation for ⊔ababaa$\underline{⊔}$:
  - ⊔ababaa$\underline{⊔}$
  - $\underline{⊔}$ababaa
  - ⊔$\underline{a}$babaa
  - ⊔a$\underline{b}$abaa // ↓b
  - ⊔$\underline{a}$babaa // ↓a
  - ⊔$\underline{b}$babaa
  - ⊔b$\underline{b}$abaa
  - ⊔b$\underline{a}$abaa
  - ⊔$\underline{b}$aabaa
  - $\underline{⊔}$baabaa // ←b, ⊔; the schema goes to the left, the head goes to the right

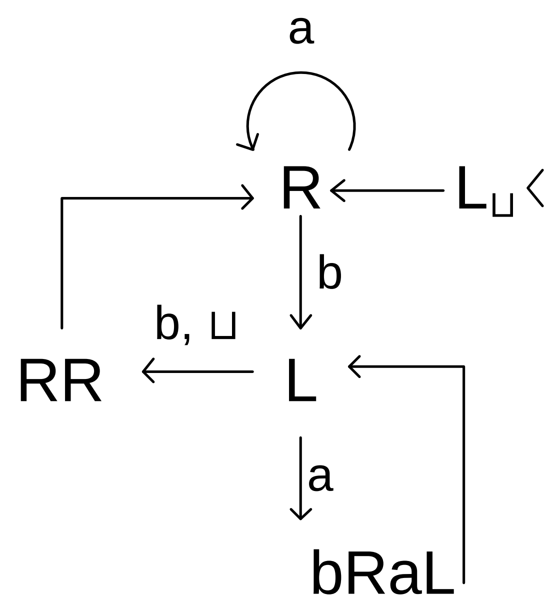
Example 16: Sort machine

- $\sqcup\underline{b}aabaa$
- $\sqcup b\underline{a}abaa$
- $\sqcup ba\underline{a}baa$
- $\sqcup baa\underline{b}aa$ // $\downarrow$b
- $\sqcup ba\underline{a}baa$ // $\downarrow$a
- $\sqcup ba\underline{b}baa$
- $\sqcup bab\underline{b}aa$
- $\sqcup bab\underline{a}aa$
- $\sqcup ba\underline{b}aaa$
- $\sqcup b\underline{a}baaa$ // $\downarrow$a
- $\sqcup b\underline{b}baaa$



Example 16: Sort machine

- ⊔bb$\underline{\text{b}}$aaa
- ⊔bb$\underline{\text{a}}$aaa
- ⊔b$\underline{\text{b}}$aaaa
- ⊔$\underline{\text{b}}$baaaa // ←b, ⊔
- ⊔b$\underline{\text{b}}$aaaa
- ⊔bb$\underline{\text{a}}$aaa
- ⊔bba$\underline{\text{a}}$aa
- ⊔bbaa$\underline{\text{a}}$a
- ⊔bbaaa$\underline{\text{a}}$
- ⊔bbaaaa$\underline{⊔}$



Example 17: Copy machine ($C^{1\to2}$)

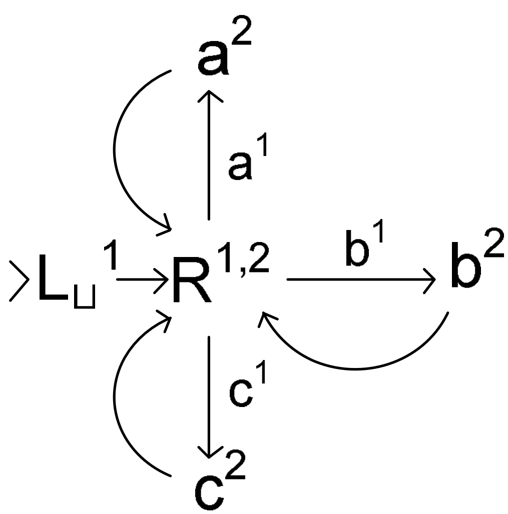
- Create a machine schema, which copies a word from the first tape to the second tape!
  - $\sum = \{\rhd, \sqcup, a, b, c\}$
  - e.g.: $\sqcup bab\underline{\sqcup}$, $\underline{\sqcup} \to \sqcup bab\underline{\sqcup}$, $\sqcup bab\underline{\sqcup}$

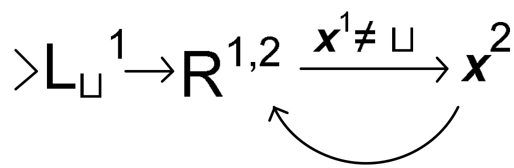
Example 17: Copy machine ($C^{1 \to 2}$)

- Create a machine schema, which copies a word from the first tape to the second tape!
  - use superscript to denote that an elementary machine works on the first or the second tape
    - e.g.: R^1 – move to the right on tape 1
    - e.g.: $L^{1,2}$ – move to the left on both tapes
  - also use superscript on the labels of the arcs
    - e.g.: b^1c^2 – follow this arc if the head of the first tape reads b, and the head of the second tape reads c

Example 17: Copy machine ($C^{1\to 2}$)

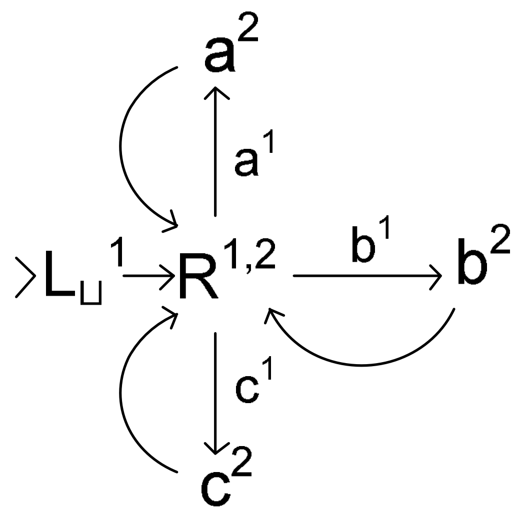
- Create a machine schema, which copies a word from the first tape to the second tape!





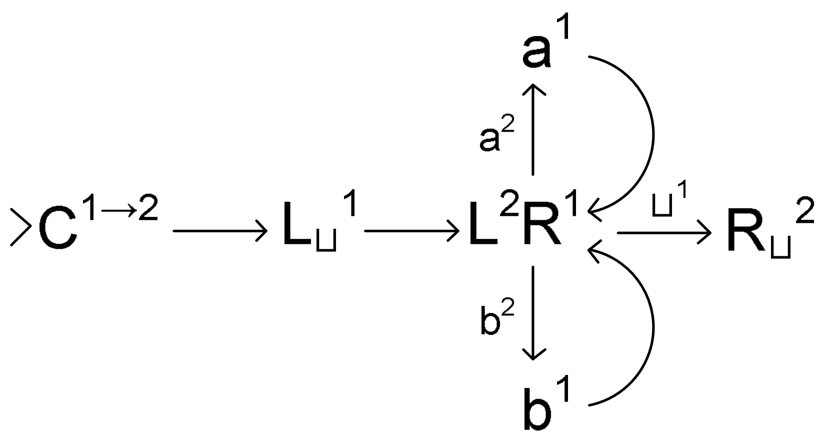
Example 17: Copy machine ($C^{1\rightarrow 2}$)

- Computation for $\sqcup abc\underline{\sqcup}, \underline{\sqcup}$:
  - $\sqcup abc\underline{\sqcup}, \underline{\sqcup}$
  - $\underline{\sqcup}abc, \underline{\sqcup}$
  - $\sqcup\underline{a}bc, \sqcup\underline{\sqcup}$
  - $\sqcup\underline{a}bc, \sqcup\underline{a}$
  - $\sqcup a\underline{b}c, \sqcup a\underline{\sqcup}$
  - $\sqcup a\underline{b}c, \sqcup a\underline{\sqcup}$
  - $\sqcup a\underline{b}c, \sqcup a\underline{b}$
  - $\sqcup ab\underline{c}, \sqcup ab\underline{\sqcup}$
  - $\sqcup ab\underline{c}, \sqcup ab\underline{c}$
  - $\sqcup abc\underline{\sqcup}, \sqcup abc\underline{\sqcup}$



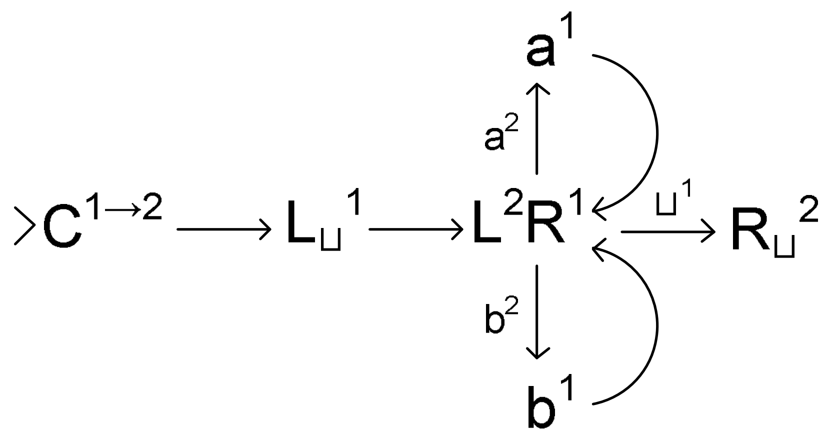
Example 18: Reverse string

- Create a machine schema, which reverses a string!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - you can use two tapes
  - e.g.: $\sqcup abab\underline{\sqcup}$, $\underline{\sqcup} \rightarrow \sqcup baba\underline{\sqcup}$, $\sqcup abab\underline{\sqcup}$



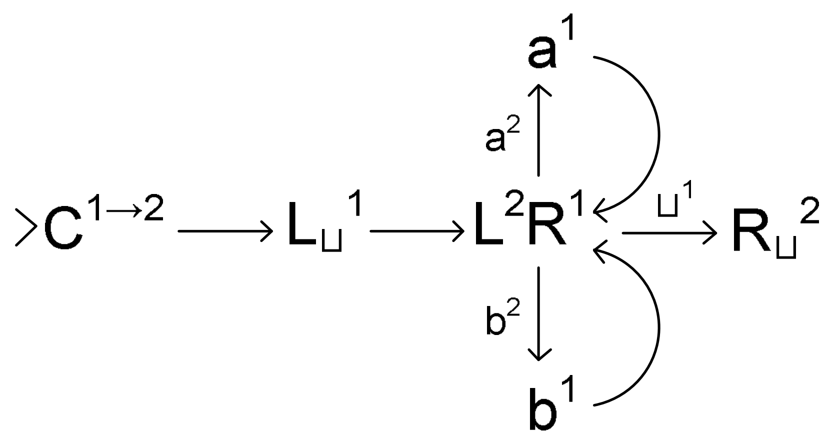
Example 18: Reverse string

- Computation for $\sqcup abab\underline{\sqcup}, \underline{\sqcup}$:
  - $\sqcup abab\underline{\sqcup}, \underline{\sqcup}$
  - $\sqcup abab\underline{\sqcup}, \sqcup abab\underline{\sqcup}$
  - $\underline{\sqcup}abab, \sqcup abab\underline{\sqcup}$
  - $\sqcup \underline{a}bab, \sqcup aba\underline{b}$
  - $\sqcup \underline{b}bab, \sqcup aba\underline{b}$
  - $\sqcup b\underline{b}ab, \sqcup ab\underline{a}b$
  - $\sqcup b\underline{a}ab, \sqcup ab\underline{a}b$



Example 18: Reverse string

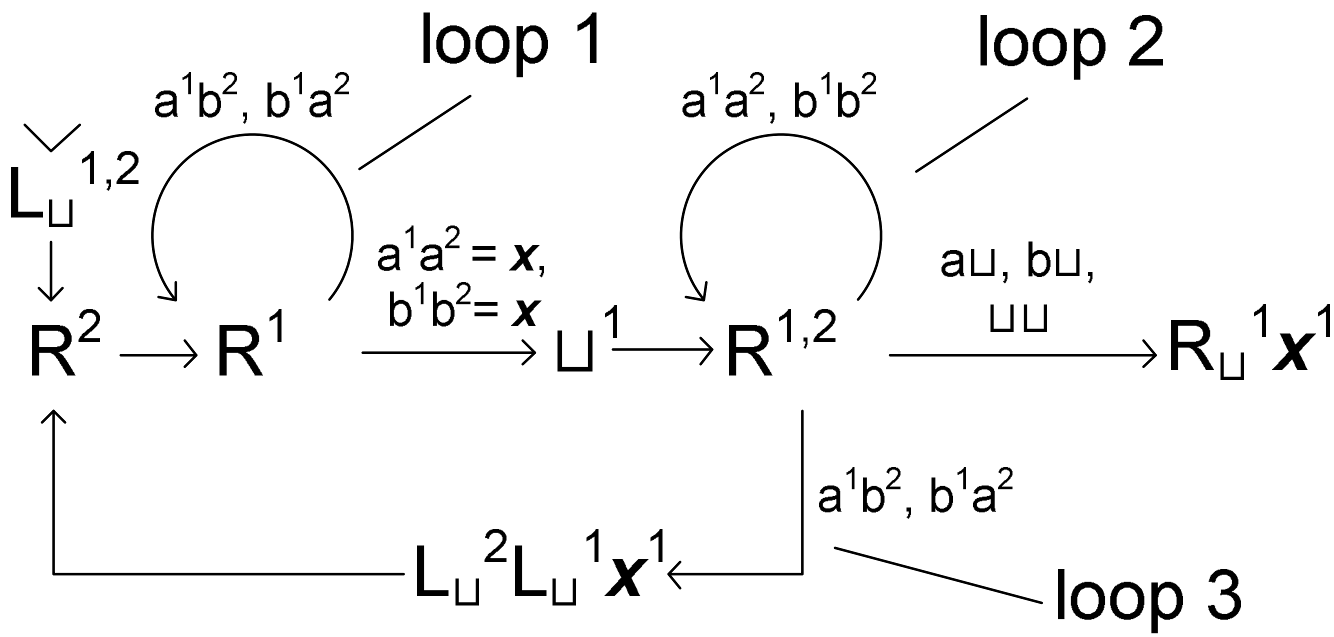
- $\sqcup ba\underline{a}b$, $\sqcup a\underline{b}ab$
- $\sqcup ba\underline{b}b$, $\sqcup a\underline{b}ab$
- $\sqcup aba\underline{b}$, $\sqcup \underline{a}bab$
- $\sqcup bab\underline{a}$, $\sqcup \underline{a}bab$
- $\sqcup baba\underline{\sqcup}$, $\underline{\sqcup}abab$
- $\sqcup baba\underline{\sqcup}$, $\sqcup abab\underline{\sqcup}$



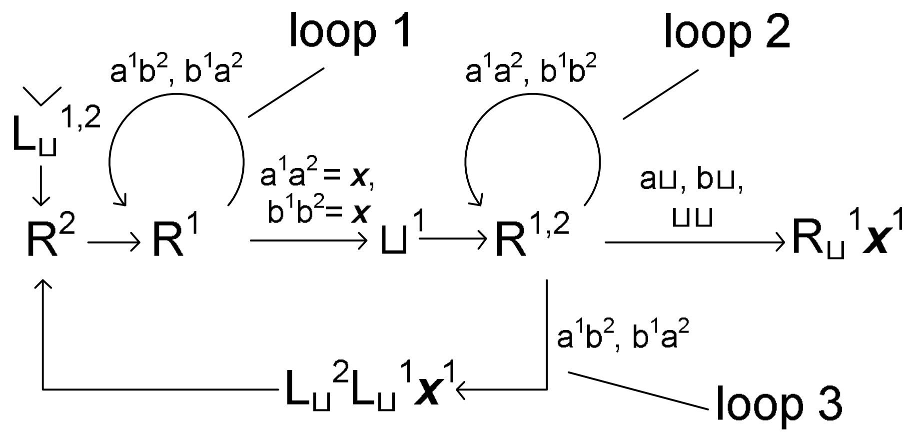
Example 19: Search substring

- Create a machine schema, which searches a substring in a string!
  - $\sum = \{\rhd, \sqcup, a, b\}$
  - tape 1 contains the string, tape 2 contains the substring
  - the string and the substring cannot be empty
  - position the head on tape 1 to the beginning of the located substring
  - e.g.: $\sqcup aabaab\underline{\sqcup}$, $\sqcup aba\underline{\sqcup} \rightarrow \sqcup a\underline{a}baab\sqcup$, $\sqcup aba\underline{\sqcup}$

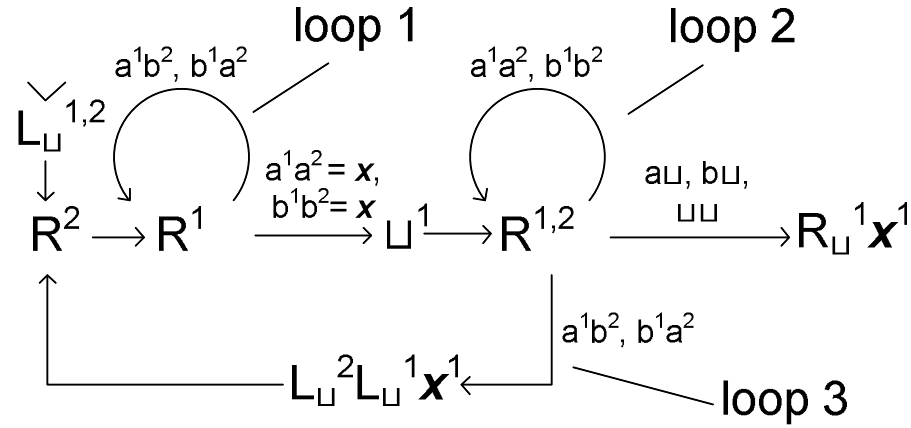
Example 19: Search substring



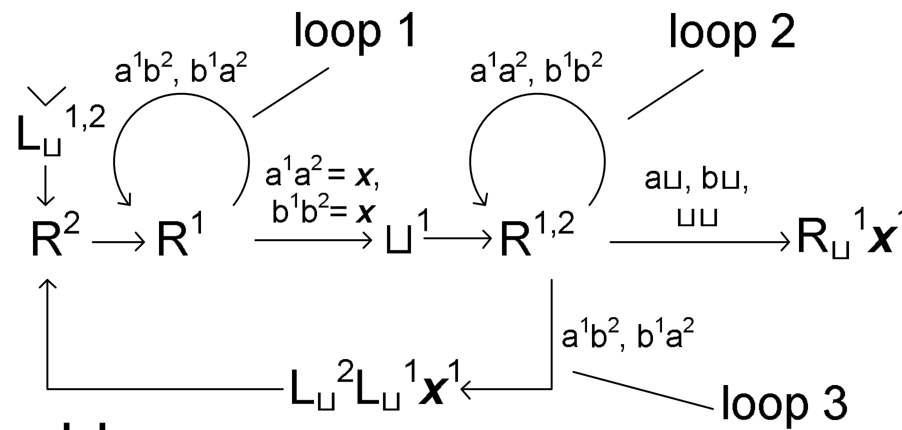
- Remark: ***x*** is a single character



- Logic of the operation:
  - loop 1 finds the first character of the substring (tape 2) in the string (tape 1)
  - this position is marked by space
  - loop 2 cycles while the string and the substring matches
  - if the end of the substring is reached then the marked position on tape 1 is restored and the machine halts
  - if there is a mismatch then loop 3 restores the marked position and the search is started after this position



- Computation:
  - $\sqcup aaaabb\underline{\sqcup}, \sqcup bb\underline{\sqcup}$
  - $\underline{\sqcup}aaaabb\sqcup, \underline{\sqcup}bb\sqcup$
  - $\underline{\sqcup}aaaabb\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup\underline{a}aaabb\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup a\underline{a}aabb\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup aa\underline{a}abb\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup aaa\underline{a}bb\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup aaaa\underline{b}b\sqcup, \sqcup\underline{b}b\sqcup$ // $\rightarrow aa, bb$
  - $\sqcup aaaa\underline{\sqcup}b\sqcup, \sqcup\underline{b}b\sqcup$
  - $\sqcup aaaa\sqcup\underline{b}\sqcup, \sqcup b\underline{b}\sqcup$
  - $\sqcup aaaa\sqcup b\underline{\sqcup}, \sqcup bb\underline{\sqcup}$ // $\rightarrow a\sqcup, b\sqcup, \sqcup\sqcup$
  - $\sqcup aaaa\underline{b}b\sqcup, \sqcup bb\underline{\sqcup}$



– $\underline{\sqcup}$aabaab⊔, $\underline{\sqcup}$aba⊔
– $\underline{\sqcup}$aabaab⊔, ⊔$\underline{a}$ba⊔
– ⊔$\underline{a}$abaab⊔, ⊔$\underline{a}$ba⊔ // →aa, bb
– ⊔$\underline{\sqcup}$abaab⊔, ⊔$\underline{a}$ba⊔
– ⊔⊔$\underline{a}$baab⊔, ⊔a$\underline{b}$a⊔ // ↓ab, ba
– ⊔$\underline{a}$abaab⊔, $\underline{\sqcup}$aba⊔
– ⊔$\underline{a}$abaab⊔, ⊔$\underline{a}$ba⊔
– ⊔a$\underline{a}$baab⊔, ⊔$\underline{a}$ba⊔ // →aa, bb
– ⊔a$\underline{\sqcup}$baab⊔, ⊔$\underline{a}$ba⊔
– ⊔a⊔$\underline{b}$aab⊔, ⊔a$\underline{b}$a⊔
– ⊔a⊔b$\underline{a}$ab⊔, ⊔ab$\underline{a}$⊔
– ⊔a⊔ba$\underline{a}$b⊔, ⊔aba$\underline{\sqcup}$ // →a⊔, b⊔, ⊔⊔
– ⊔a$\underline{a}$baab⊔, ⊔aba$\underline{\sqcup}$

Example 19: Search substring

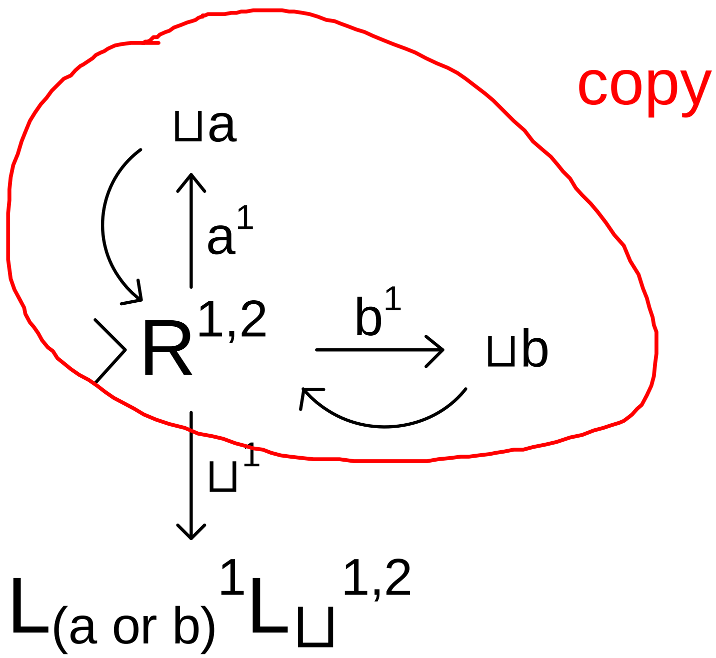
- Homework:
  - give the computation for:
    - bbaababa, bab
    - bbaababa, abba
      - how does the machine stop if the substring is not in the string?
  - give a similar machine with $\Sigma = \{\rhd, \sqcup, a, b, c\}$

Example 20: Split string

- Create a machine schema, which can split a string into 2 tapes based on the head position!
  - ∑ = {⊳, ⊔, a, b}
  - the input must be at least 2 character long
  - after the machine finished
    - the first tape should contain the input from the first character until the head (including the head position)
    - the second tape should contain the rest
  - e.g.: ⊔a$\underline{\text{a}}$bab, $\underline{⊔}$ → ⊔aa$\underline{⊔}$, ⊔bab$\underline{⊔}$

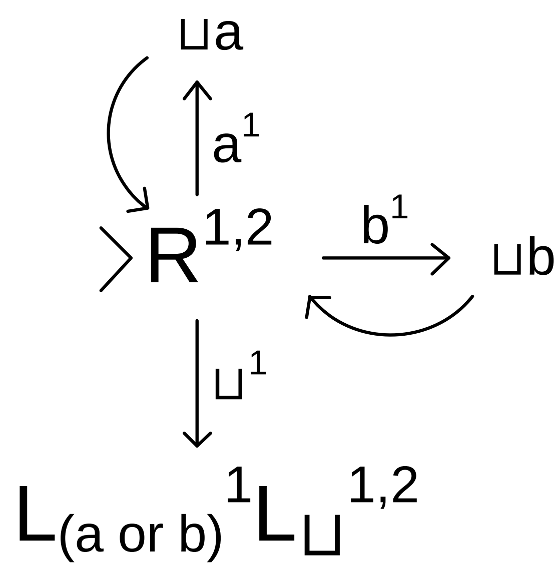
Example 20: Split string

- Create a machine schema, which can cut a string to 2 tapes!



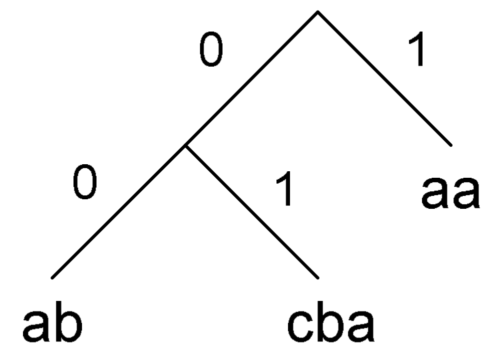
Example 20: Split string

- Computation:
  - $\sqcup a\underline{a}bab, \underline{\sqcup}$
  - $\sqcup aa\underline{b}ab, \sqcup\underline{\sqcup}$ // $\rightarrow b^1$
  - $\sqcup aa\underline{\sqcup}ab, \sqcup\underline{b}$
  - $\sqcup aa\sqcup\underline{a}b, \sqcup b\underline{\sqcup}$ // $\uparrow a^1$
  - $\sqcup aa\sqcup\underline{\sqcup}b, \sqcup b\underline{a}$
  - $\sqcup aa\sqcup\sqcup\underline{b}, \sqcup ba\underline{\sqcup}$ // $\rightarrow b^1$
  - $\sqcup aa\sqcup\sqcup\underline{\sqcup}, \sqcup ba\underline{b}$
  - $\sqcup aa\sqcup\sqcup\sqcup\underline{\sqcup}, \sqcup bab\underline{\sqcup}$ // $\downarrow\sqcup^1$
  - $\sqcup a\underline{a}, \sqcup bab\underline{\sqcup}$
  - $\underline{\sqcup}aa, \underline{\sqcup}bab\sqcup$



Example 21: Huffman decoding

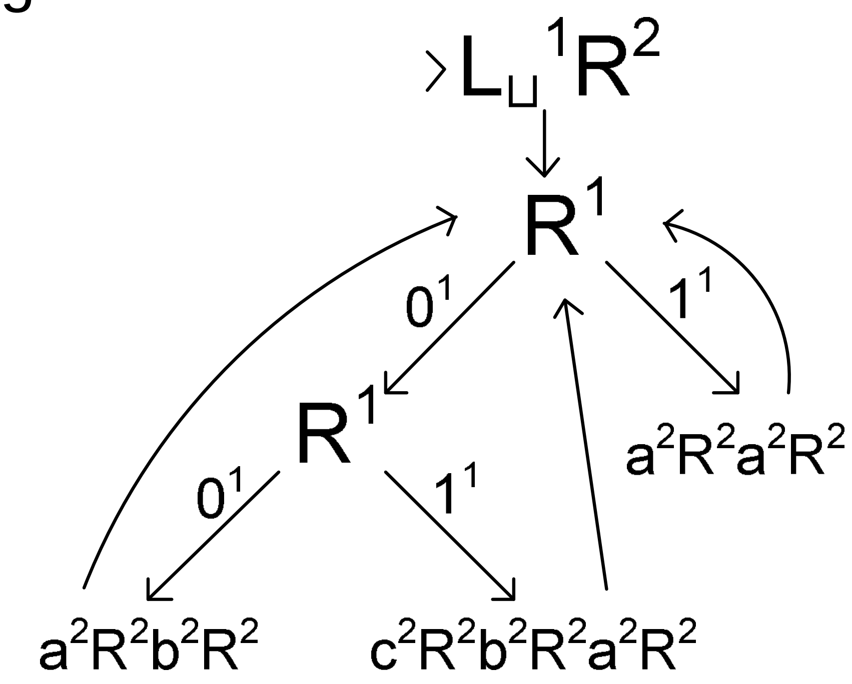
- Create a machine schema, which can decode a Huffman encoded string!
  - tape 1 contains the coded string, tape 2 will contain the decoded string
  - decoding: change the code to string
    - 1 $\rightarrow$ aa
    - 00 $\rightarrow$ ab
    - 01 $\rightarrow$ cba



  - the creation of the tree is not detailed here
  - e.g.: $\sqcup 101001\underline{\sqcup}, \sqcup\underline{\sqcup} \rightarrow \sqcup 101001\underline{\sqcup}, \sqcup \mathrm{aacbaabaa}\underline{\sqcup}$

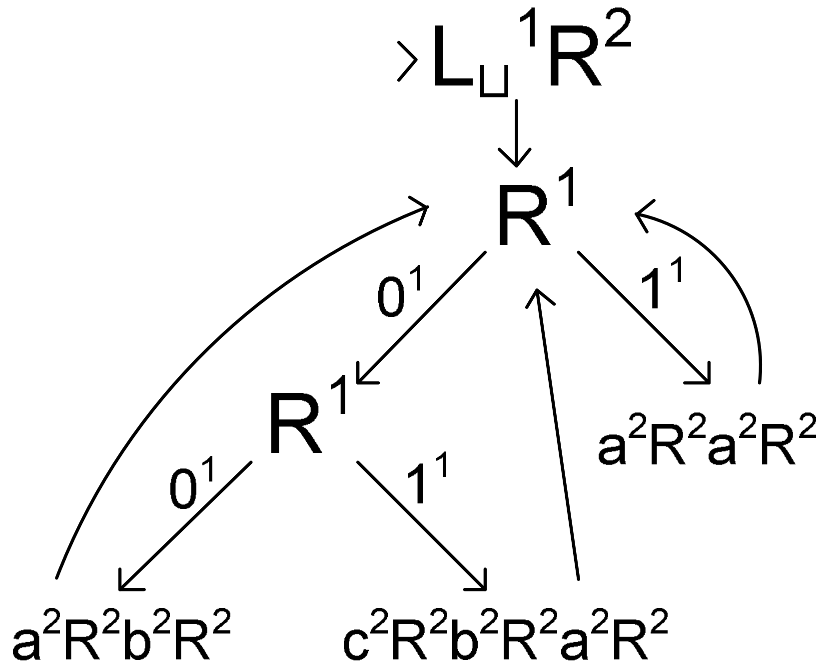
Example 21: Huffman decoding

- Create a machine schema, which can decode a Huffman encoded string!



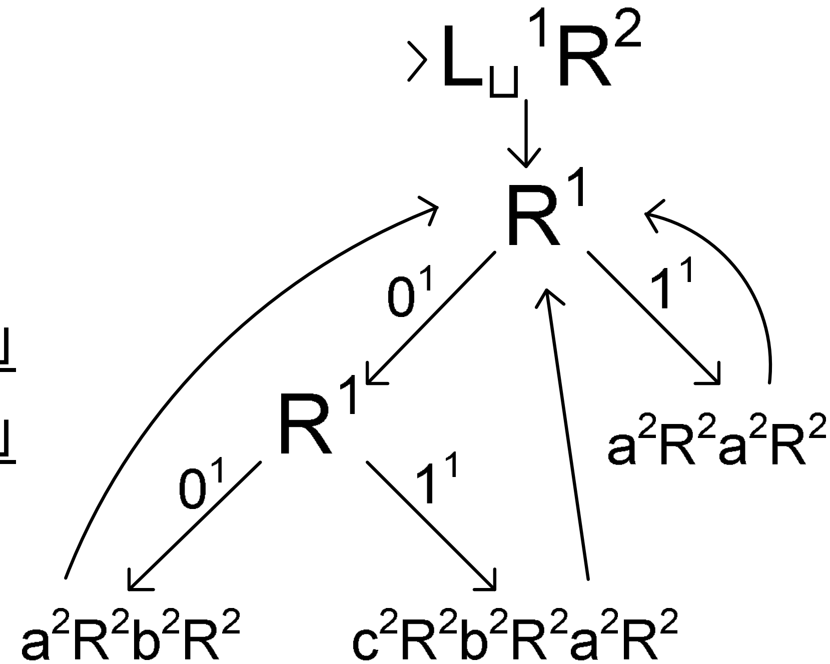
Example 21: Huffman decoding

- Computation:
  - $\sqcup 101001 \underline{\sqcup}, \sqcup \underline{\sqcup}$
  - $\underline{\sqcup} 101001 \sqcup, \sqcup \underline{\sqcup}$
  - $\sqcup \underline{1} 01001 \sqcup, \sqcup aa \underline{\sqcup}$
  - $\sqcup 1 \underline{0} 1001 \sqcup, \sqcup aa \underline{\sqcup}$
  - $\sqcup 10 \underline{1} 001 \sqcup, \sqcup aa \underline{\sqcup}$
  - $\sqcup 10 \underline{1} 001 \sqcup, \sqcup aacba \underline{\sqcup}$



Example 21: Huffman decoding

- $\sqcup 101\underline{0}01\sqcup$, $\sqcup \mathrm{aacba}\underline{\sqcup}$
- $\sqcup 1010\underline{0}1\sqcup$, $\sqcup \mathrm{aacba}\underline{\sqcup}$
- $\sqcup 1010\underline{0}1\sqcup$, $\sqcup \mathrm{aacbaab}\underline{\sqcup}$
- $\sqcup 10100\underline{1}\sqcup$, $\sqcup \mathrm{aacbaab}\underline{\sqcup}$
- $\sqcup 10100\underline{1}\sqcup$, $\sqcup \mathrm{aacbaabaa}\underline{\sqcup}$
- $\sqcup 101001\underline{\sqcup}$, $\sqcup \mathrm{aacbaabaa}\underline{\sqcup}$

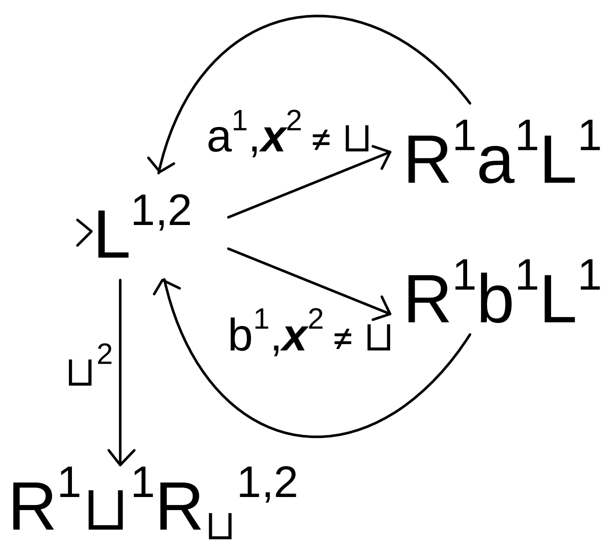


Example 22: Shifting machine ($Sh_{\rightarrow}^{1,2}$)

- Construct a machine schema, which can shift to the right the last n character of a string in tape 1!
  - $\Sigma_1 \in \{a, b\}$
  - n is determined by the length of an other string in tape 2
  - the newly inserted character is the blank
  - e.g.: $\sqcup aaaba\underline{\sqcup}, \sqcup{**}\underline{\sqcup} \rightarrow \sqcup aaa\sqcup ba\underline{\sqcup}, \sqcup{**}\underline{\sqcup}$
    - n = 2

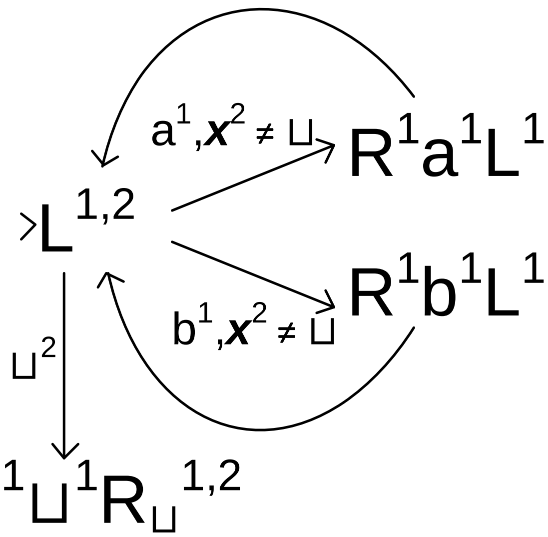
Example 22: Shifting machine ($Sh_{\rightarrow}^{1,2}$)

- Construct a machine schema, which can shift to the right the last n character of a string in tape 1!



Example 22: Shifting machine ($Sh_{\rightarrow}^{1,2}$)

- Computation:
  - $\sqcup aba\underline{\sqcup}$, $\sqcup{*}{*}\underline{\sqcup}$
  - $\sqcup ab\underline{a}$, $\sqcup{*}\underline{*}\sqcup$ // $\rightarrow a^1$
  - $\sqcup ab\underline{a}a$, $\sqcup{*}\underline{*}\sqcup$
  - $\sqcup a\underline{b}aa$, $\sqcup\underline{*}{*}\sqcup$ // $\rightarrow b^1$
  - $\sqcup a\underline{b}ba$, $\sqcup\underline{*}{*}\sqcup$
  - $\sqcup\underline{a}bba$, $\underline{\sqcup}{*}{*}\sqcup$ // $\downarrow\sqcup^2$
  - $\sqcup a\sqcup ba\underline{\sqcup}$, $\sqcup{*}{*}\underline{\sqcup}$



Example 23: Compare machine (T_{CMP})

- Create a machine schema, which can compare 2 strings!
  - $\Sigma_1 \in \{a, b\}$
  - if the two strings match then position the heads after the strings on the spaces, otherwise the head should be somewhere on the strings
    - a machine schema can check if the heads are on spaces ($\rightarrow\sqcup\sqcup$)
  - e.g.: $\sqcup aba\underline{\sqcup}$, $\sqcup aba\underline{\sqcup} \rightarrow \sqcup aba\underline{\sqcup}$, $\sqcup aba\underline{\sqcup}$

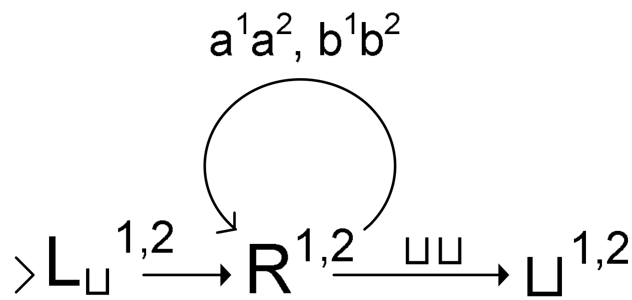
Example 23: Compare machine (T_{CMP})

- Create a machine schema, which can compare 2 strings!
  - the arc "$\rightarrow\sqcup\sqcup$" is there to sign that, if we go through this arc then the two string equal

$$
\begin{tikzcd}
{>}L_{\sqcup}^{1,2} \arrow[r] & R^{1,2} \arrow[loop above, "a^1a^2,\ b^1b^2"] \arrow[r, "\sqcup\sqcup"] & \sqcup^{1,2}
\end{tikzcd}
$$

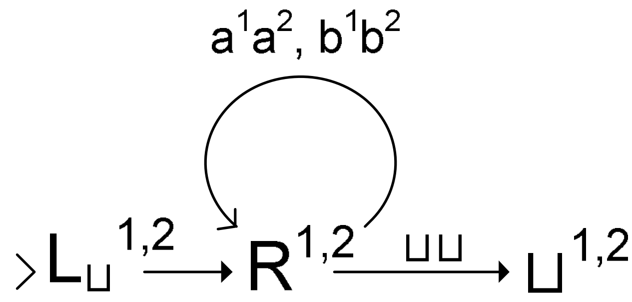
Example 23: Compare machine (T_{CMP})

- Computation:
  - $\sqcup aba\underline{\sqcup}, \sqcup aba\underline{\sqcup}$
  - $\underline{\sqcup}aba\sqcup, \underline{\sqcup}aba\sqcup$
  - $\sqcup\underline{a}ba\sqcup, \sqcup\underline{a}ba\sqcup$
  - $\sqcup a\underline{b}a\sqcup, \sqcup a\underline{b}a\sqcup$
  - $\sqcup ab\underline{a}\sqcup, \sqcup ab\underline{a}\sqcup$
  - $\sqcup aba\underline{\sqcup}, \sqcup aba\underline{\sqcup}$ // $\rightarrow\sqcup\sqcup$
  - $\sqcup aba\underline{\sqcup}, \sqcup aba\underline{\sqcup}$



Example 23: Compare machine (T_{CMP})

- Computation:
  - $\sqcup ab\underline{\sqcup}$, $\sqcup aa\underline{\sqcup}$
  - $\underline{\sqcup}ab\sqcup$, $\underline{\sqcup}aa\sqcup$
  - $\sqcup\underline{a}b\sqcup$, $\sqcup\underline{a}a\sqcup$
  - $\sqcup a\underline{b}\sqcup$, $\sqcup a\underline{a}\sqcup$ // string rejected



Example 24: Accept string zzz

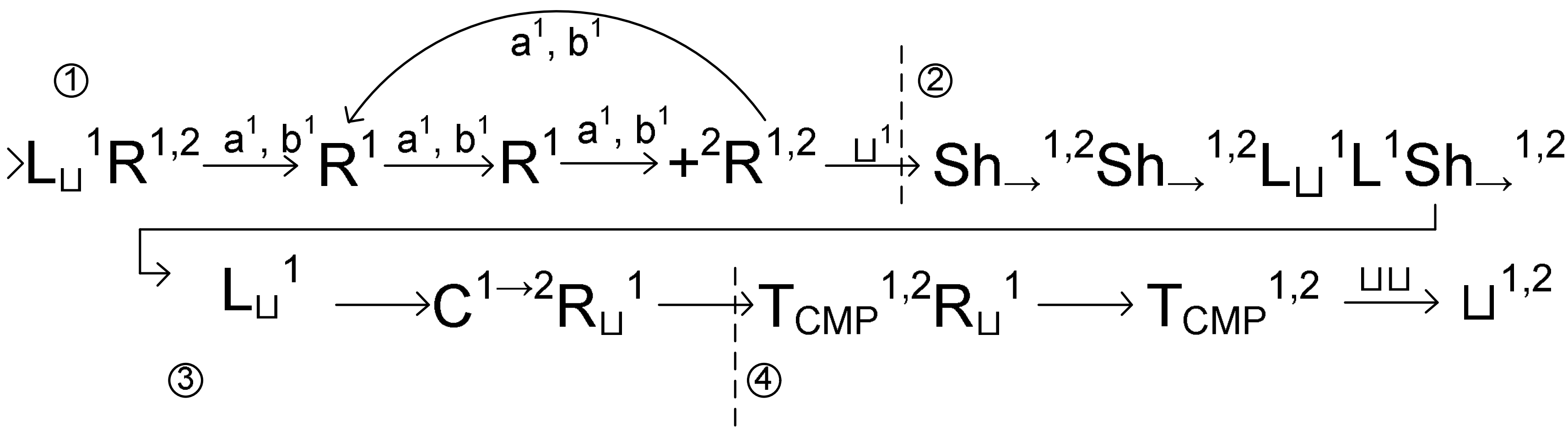
- Create a machine schema, which can accept $L(M) = \{w \in \{a, b\}^* \mid w = zzz\}$!
  - if the input is in the zzz form then position the head after the input on the space
    - a machine schema can check if the head is on space ($\rightarrow\sqcup$)
  - you can modify the input and you can use a second tape
  - e.g.: $\sqcup ababab\underline{\sqcup}$, $\underline{\sqcup} \rightarrow \sqcup ab \sqcup ab \sqcup ab\underline{\sqcup}$, $\sqcup{+}{+}\sqcup ab\underline{\sqcup}$

Example 24: Accept string zzz

– beside the usual machines, the following machines can be also used
  - T_{CMP}: compare two strings on two tapes
    – ⊔aba$\underline{⊔}$, ⊔aba$\underline{⊔}$ → ⊔aba$\underline{⊔}$, ⊔aba$\underline{⊔}$
  - $C^{1\to 2}$: copy a string from tape 1 to tape 2
    – ⊔bab$\underline{⊔}$, $\underline{⊔}$ → ⊔bab$\underline{⊔}$, ⊔bab$\underline{⊔}$
  - $Sh_{\to}^{1,2}$: shifts a string on tape 1 based on the length of another string on tape 2
    – ⊔aaaba$\underline{⊔}$, ⊔**$\underline{⊔}$ → ⊔aaa⊔ba$\underline{⊔}$, ⊔**$\underline{⊔}$

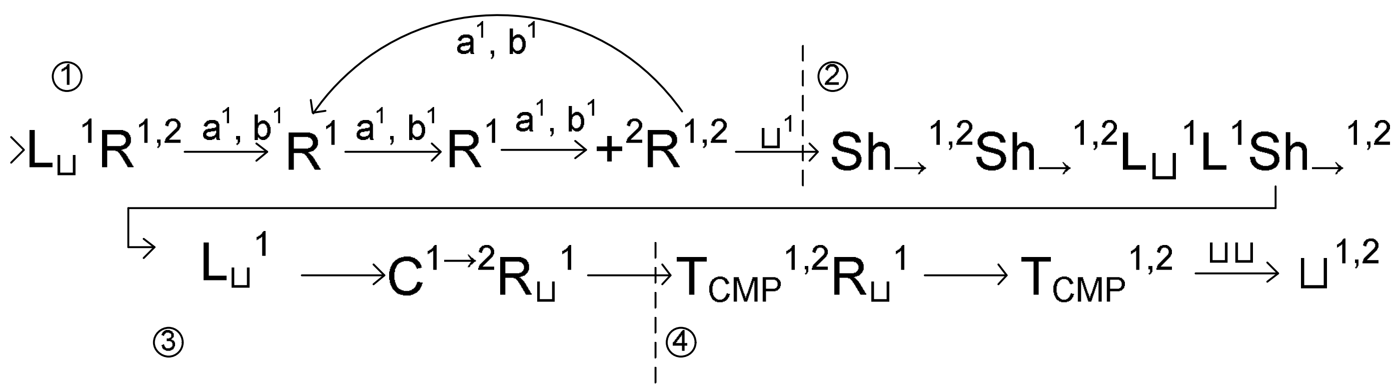
Example 24: Accept string zzz

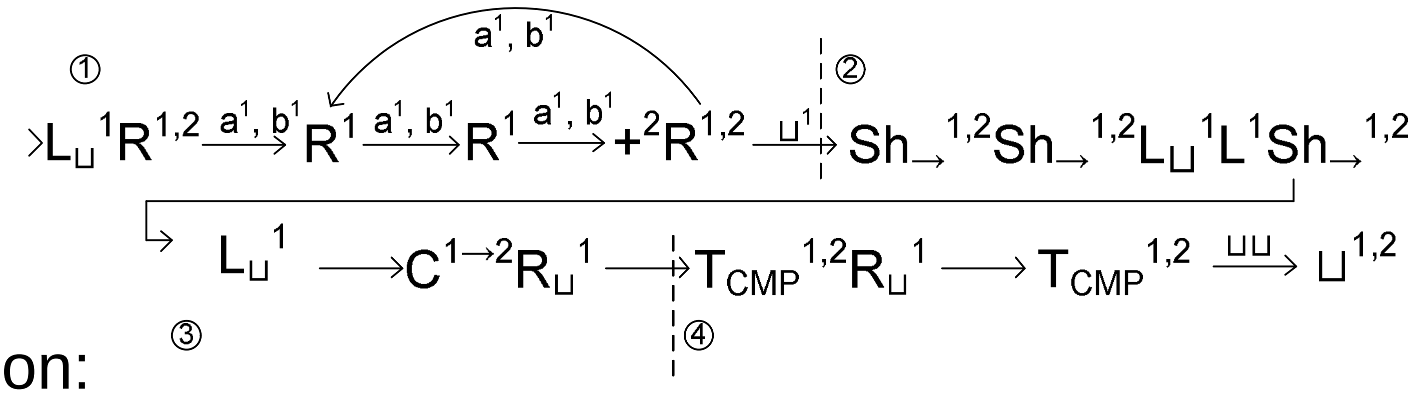
- Create a machine schema, which can accept
  L(M) = {w ∈ {a, b}* | w = zzz} !



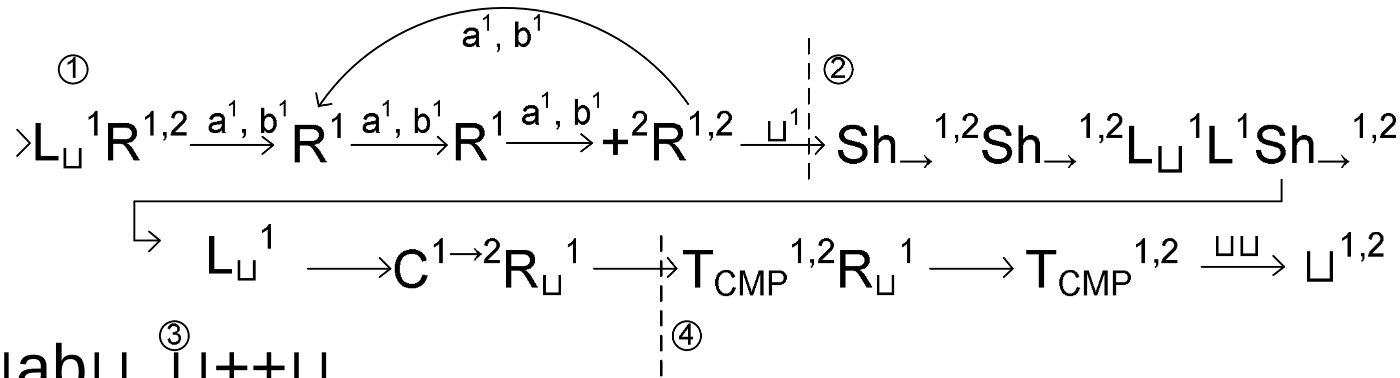
Example 24: Accept string zzz

- Logic of the operation:
  - block 1 write a '+' on tape 2 after 3 right steps on tape 1
    - computation halts if the input's length cannot be divided by 3
  - block 2 splits the string to ⊔z⊔z⊔z⊔ using $Sh_{\rightarrow}^{1,2}$
  - block 3 copies the first substring to the second tape
  - block 4 compares the second and the third substring with the first





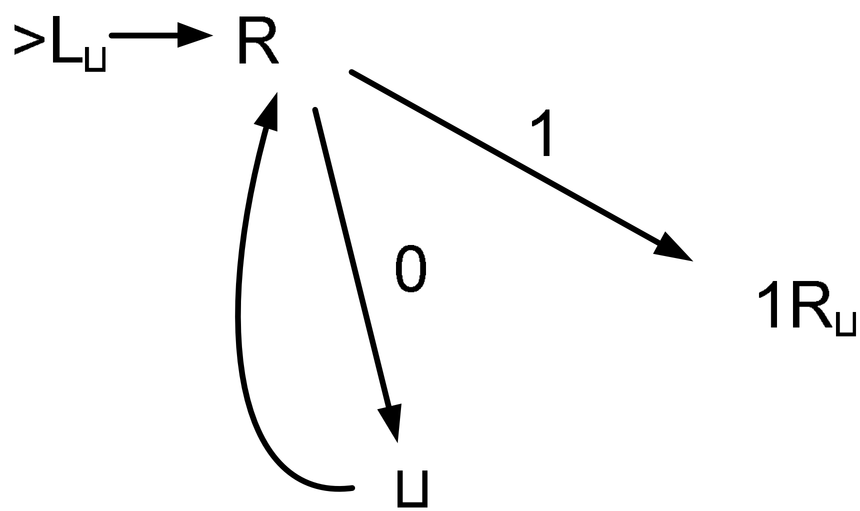
- Computation:
  - $\sqcup ababab\underline{\sqcup}, \underline{\sqcup}$
  - $\sqcup \underline{a}babab, \sqcup\underline{\sqcup}$
  - $\sqcup a\underline{b}abab, \sqcup\underline{\sqcup}$
  - $\sqcup ab\underline{a}bab, \sqcup\underline{\sqcup}$
  - $\sqcup aba\underline{b}ab, \sqcup+\underline{\sqcup}$
  - $\sqcup abab\underline{a}b, \sqcup+\underline{\sqcup}$
  - $\sqcup ababa\underline{b}, \sqcup+\underline{\sqcup}$
  - $\sqcup ababab\underline{\sqcup}, \sqcup++\underline{\sqcup}$ // block 1 ends



- ⊔abab⊔ab$\underline{\sqcup}$, ⊔++$\underline{\sqcup}$
- ⊔abab⊔⊔ab$\underline{\sqcup}$, ⊔++$\underline{\sqcup}$
- ⊔abab⊔$\underline{\sqcup}$ab⊔, ⊔++$\underline{\sqcup}$
- ⊔abab$\underline{\sqcup}$⊔ab⊔, ⊔++$\underline{\sqcup}$
- ⊔ab⊔ab$\underline{\sqcup}$ab⊔, ⊔++$\underline{\sqcup}$ // block 2 ends
- ⊔ab$\underline{\sqcup}$ab⊔ab⊔, ⊔++$\underline{\sqcup}$
- ⊔ab$\underline{\sqcup}$ab⊔ab⊔, ⊔++⊔ab$\underline{\sqcup}$
- ⊔ab⊔ab$\underline{\sqcup}$ab⊔, ⊔++⊔ab$\underline{\sqcup}$ // block 3 ends
- ⊔ab⊔ab$\underline{\sqcup}$ab⊔, ⊔++⊔ab$\underline{\sqcup}$
- ⊔ab⊔ab⊔ab$\underline{\sqcup}$, ⊔++⊔ab$\underline{\sqcup}$
- ⊔ab⊔ab⊔ab$\underline{\sqcup}$, ⊔++⊔ab$\underline{\sqcup}$ // block 4 ends

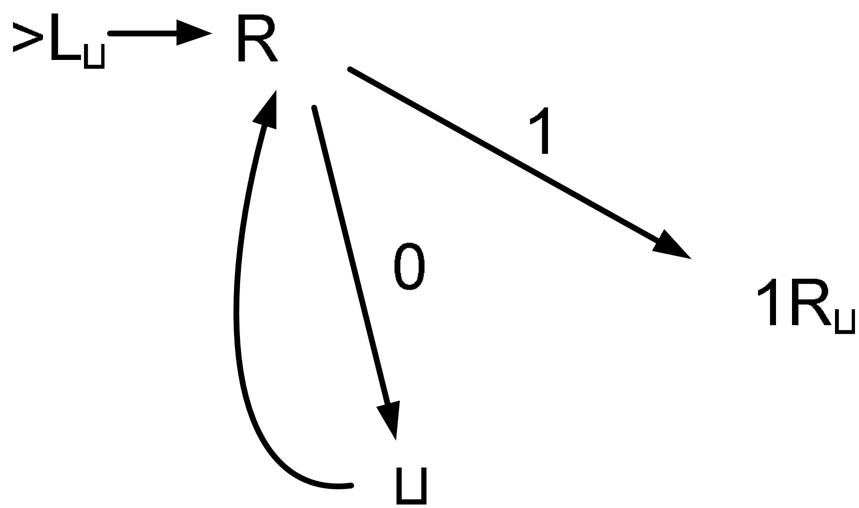
Example 25

- Create a machine schema, which erases all "0"-s from the left side of a binary number!
  - $\Sigma_1 \in \{0, 1\}$
  - e.g.: $\sqcup 000111010\underline{\sqcup} \rightarrow \sqcup\sqcup\sqcup\sqcup 111010\underline{\sqcup}$



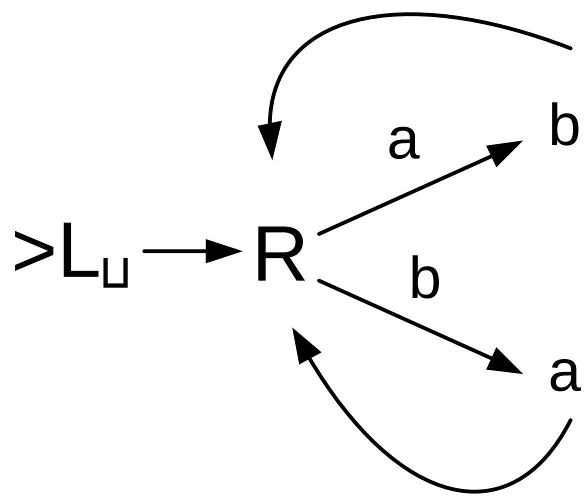
Example 25

- Computation:
  - $\sqcup 000101110\underline{\sqcup}$
  - $\underline{\sqcup}000101110$
  - $\sqcup\underline{0}00101110$
  - $\sqcup\underline{\sqcup}00101110$
  - $\sqcup\sqcup\underline{0}0101110$
  - $\sqcup\sqcup\underline{\sqcup}0101110$
  - $\sqcup\sqcup\sqcup\underline{0}101110$
  - $\sqcup\sqcup\sqcup\underline{\sqcup}101110$
  - $\sqcup\sqcup\sqcup\sqcup 101110\underline{\sqcup}$



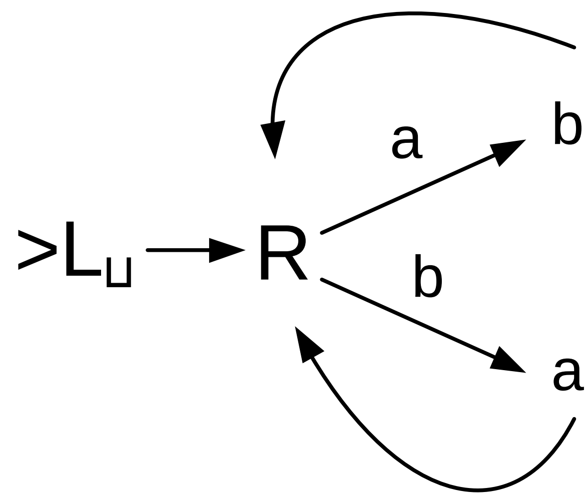
Example 26

- Create a machine schema, which switches all "a" with "b" and vice versa in a string!
  - $\Sigma_1 \in \{a, b\}$
  - e.g.: $\sqcup abab\underline{\sqcup} \rightarrow \sqcup baba\underline{\sqcup}$



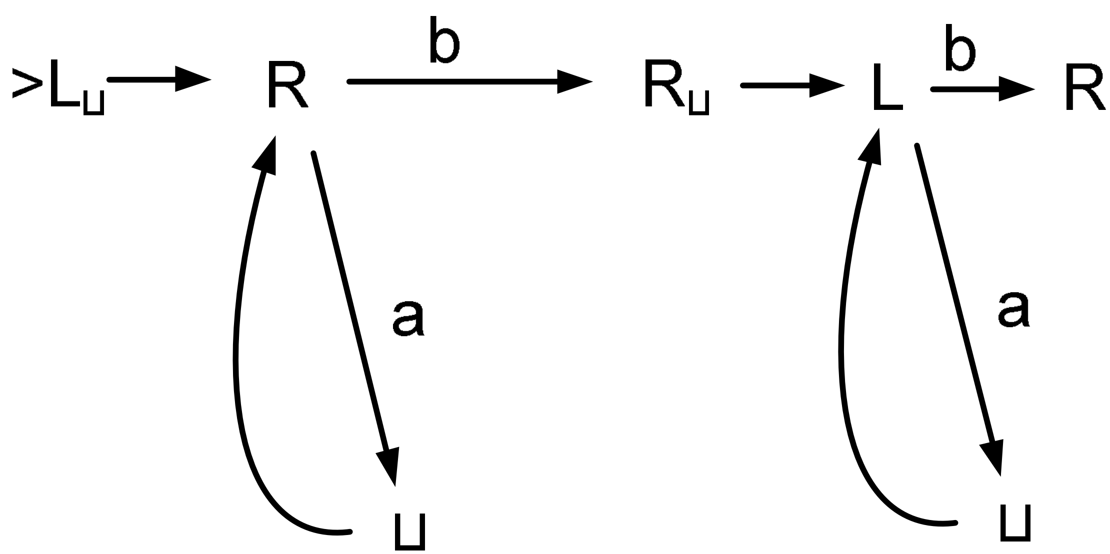
Example 26

- Computation:
  - $\sqcup abbb\underline{\sqcup}$
  - $\underline{\sqcup}abbb$
  - $\sqcup\underline{a}bbb$
  - $\sqcup\underline{b}bbb$
  - $\sqcup b\underline{b}bb$
  - $\sqcup b\underline{a}bb$
  - $\sqcup ba\underline{b}b$
  - $\sqcup ba\underline{a}b$
  - $\sqcup baa\underline{b}$
  - $\sqcup baa\underline{a}$
  - $\sqcup baaa\underline{\sqcup}$



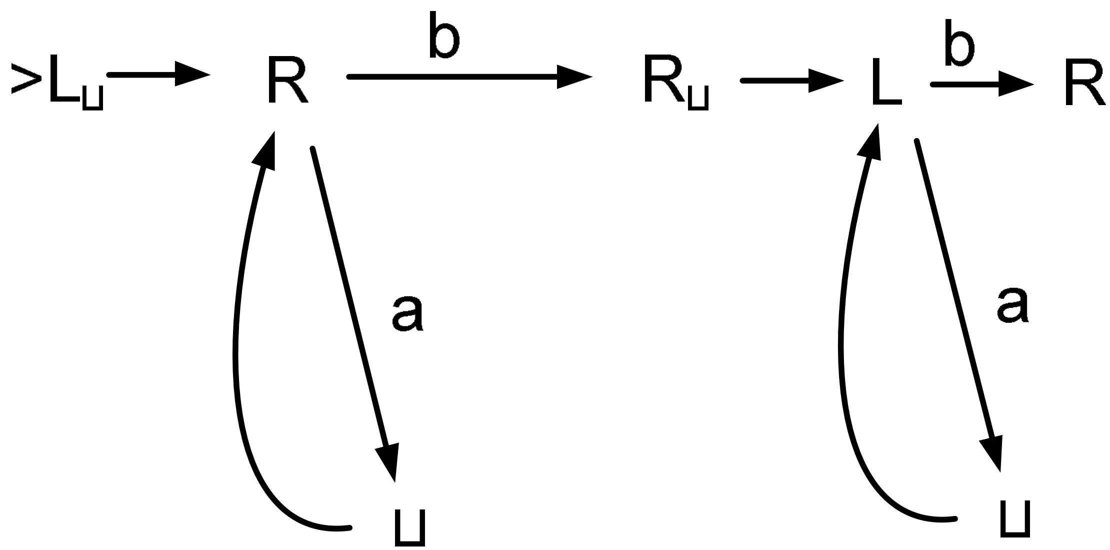
Example 27

- Create a machine schema, which erases all 'a' from the beginning and the end of the input string!
  - $\Sigma_1 \in \{a, b\}$
  - e.g.: ⊔abba<u>⊔</u> → ⊔⊔bb<u>⊔</u>



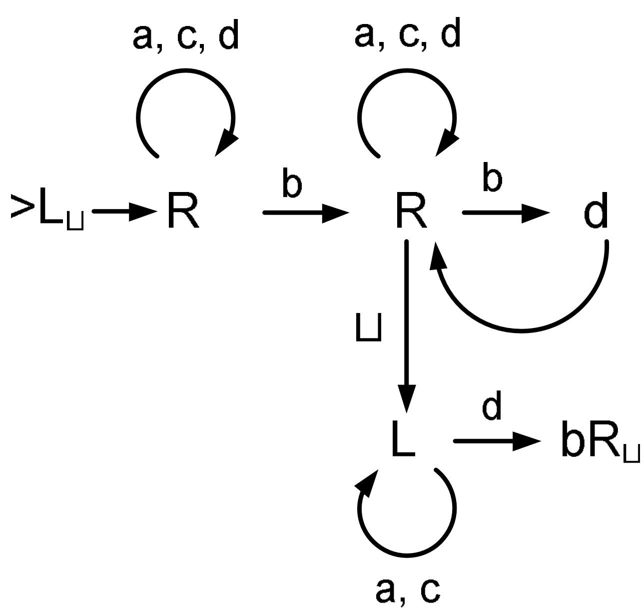
Example 27

- Computation:
  - ⊔abba$\underline{\sqcup}$
  - $\underline{\sqcup}$abba
  - ⊔$\underline{a}$bba
  - ⊔$\underline{\sqcup}$bba
  - ⊔⊔$\underline{b}$ba
  - ⊔⊔bba$\underline{\sqcup}$
  - ⊔⊔bb$\underline{a}$
  - ⊔⊔bb$\underline{\sqcup}$
  - ⊔⊔b$\underline{b}$
  - ⊔⊔bb$\underline{\sqcup}$



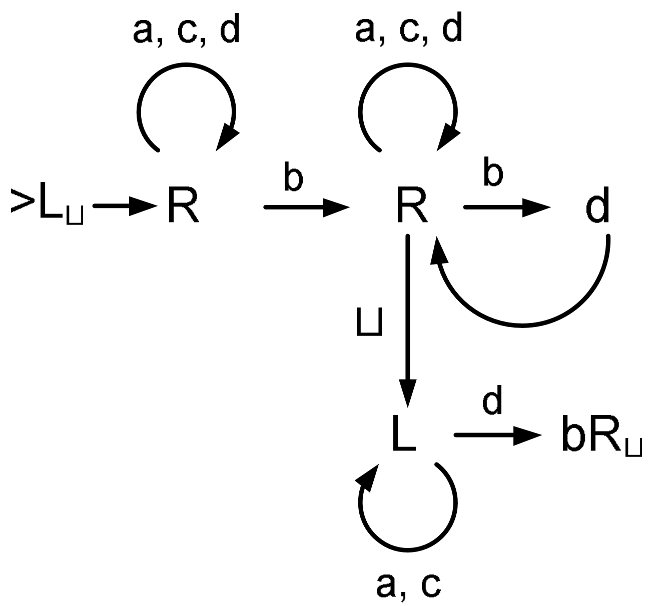
Example 28

- Create a machine schema, which changes all 'b' to 'd' in the input string, except the first and the last ones!
  – $\Sigma_1 \in \{a, b, c, d\}$
  – e.g.: ⊔abcbbbc<u>⊔</u> → ⊔abcddbc<u>⊔</u>



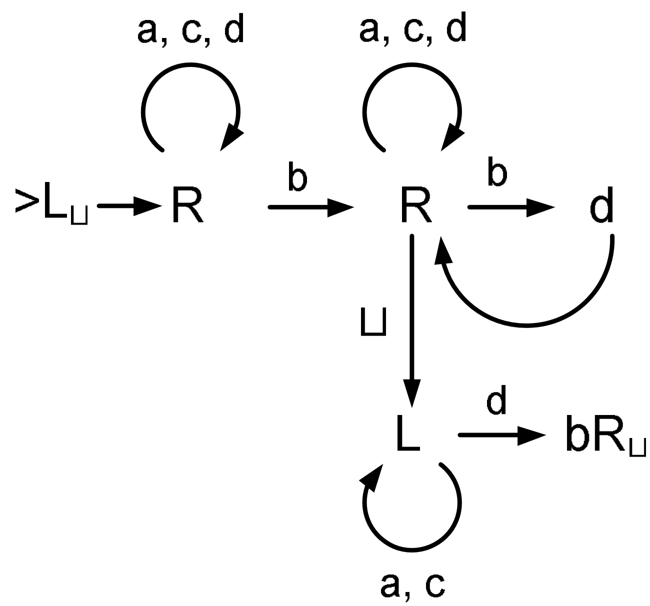
Example 28

- Computation:
  - ⊔abcbbbc<u>⊔</u>
  - <u>⊔</u>abcbbbc
  - ⊔<u>a</u>bcbbbc
  - ⊔a<u>b</u>cbbbc
  - ⊔ab<u>c</u>bbbc
  - ⊔abc<u>b</u>bbc
  - ⊔abc<u>d</u>bbc
  - ⊔abcd<u>b</u>bc
  - ⊔abcd<u>d</u>bc



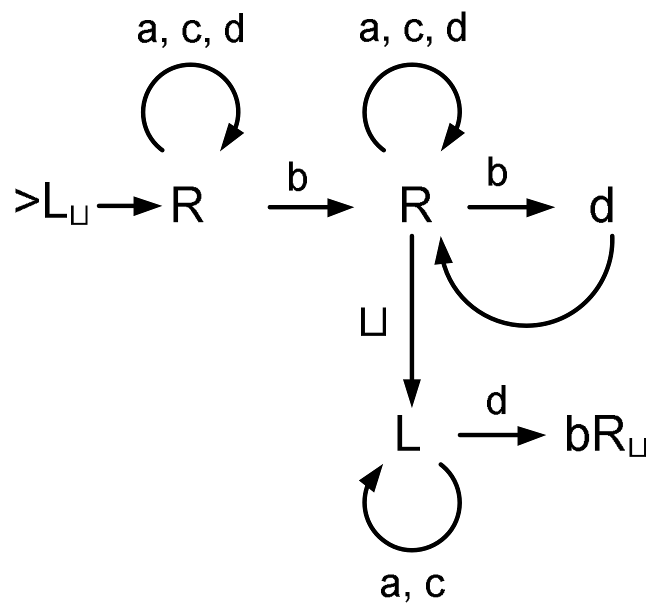
Example 28

- ⊔abcdd<u>b</u>c
- ⊔abcdd<u>d</u>c
- ⊔abcddd<u>c</u>
- ⊔abcdddc<u>⊔</u>
- ⊔abcddd<u>c</u>
- ⊔abcdd<u>d</u>c
- ⊔abcdd<u>b</u>c
- ⊔abcddbc<u>⊔</u>



Example 28

- Is this machine correct always?
  - not if the input contains d
  - e.g.: adda → adba

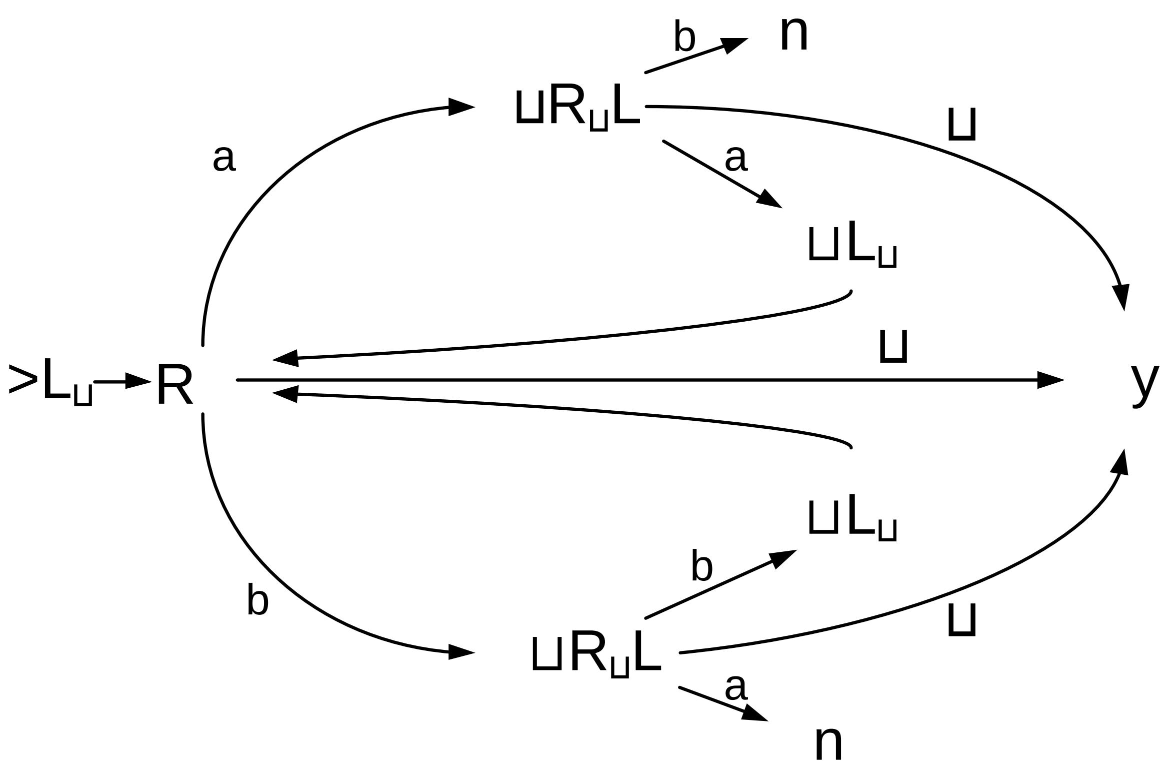


Example 29

- Create a machine schema, which decides
  $L = \{w \in \Sigma^* \mid w = w^R\}$!
  - $\Sigma_1 \in \{a, b\}$
  - if the decision is yes then perform the y machine
  - if the decision is no then perform the n machine
  - the input can be modified
  - e.g.: $\sqcup abbbba\underline{\sqcup} \rightarrow y$
    - y is the halting machine

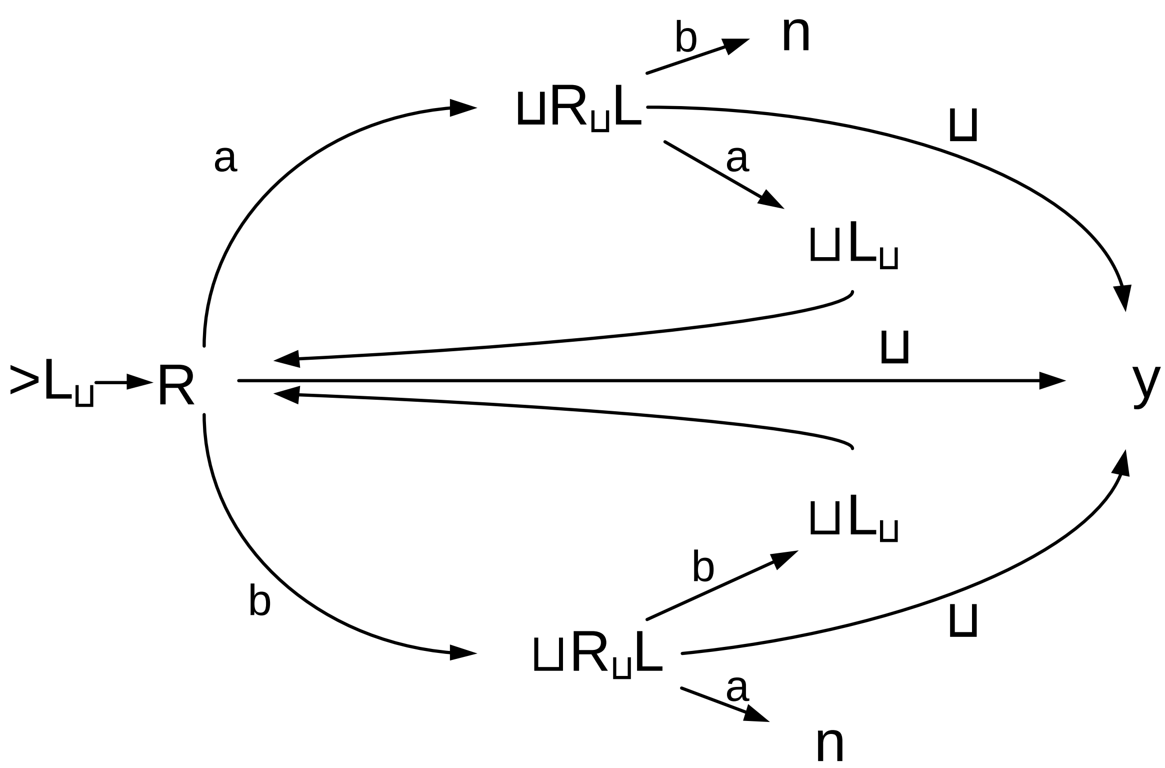
Example 29

- Create a machine schema, which decides
  $L = \{w \in \Sigma^* \mid w = w^R\}$!



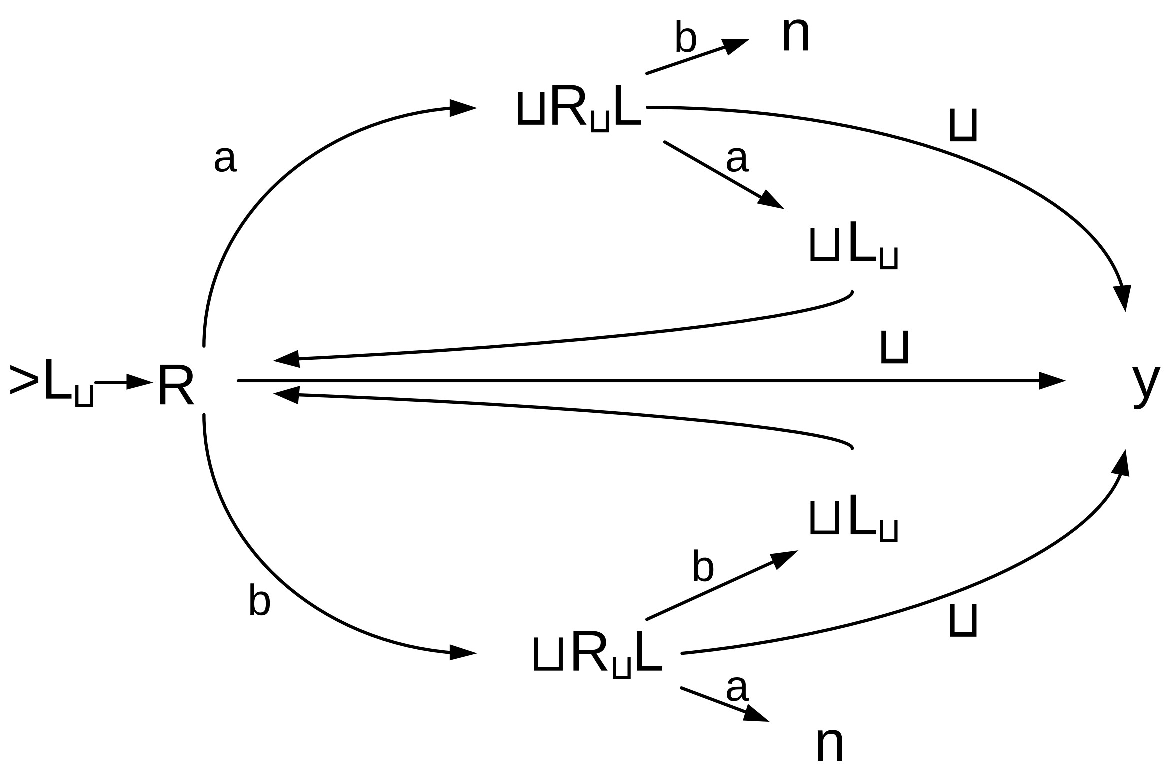
Example 29

- Computation:
  - $\sqcup abba\underline{\sqcup}$
  - $\underline{\sqcup}abba$
  - $\sqcup\underline{a}bba$
  - $\sqcup\underline{\sqcup}bba$
  - $\sqcup\sqcup bba\underline{\sqcup}$
  - $\sqcup\sqcup bb\underline{a}$
  - $\sqcup\sqcup bb\underline{\sqcup}$
  - $\sqcup\underline{\sqcup}bb$
  - $\sqcup\sqcup\underline{b}b$



Example 29

- ⊔⊔<u>⊔</u>b
- ⊔⊔⊔b<u>⊔</u>
- ⊔⊔⊔<u>b</u>
- ⊔⊔⊔<u>⊔</u>
- ⊔⊔<u>⊔</u>
- ⊔⊔⊔<u>⊔</u> // → ⊔
- ⊔⊔⊔<u>⊔</u>

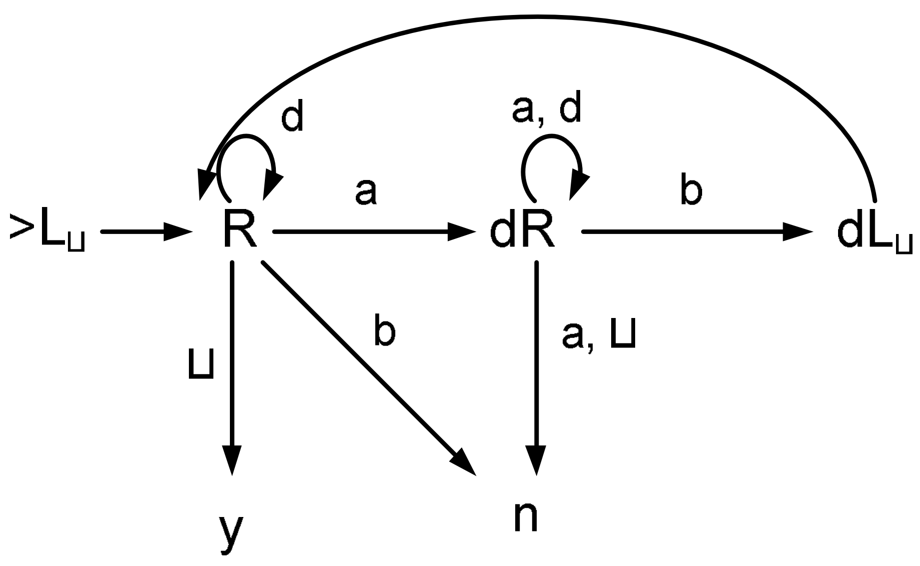


Example 30

- Create a machine schema, which decides
  $L = \{w \in \{a, b\}^* \mid w = a^n b^n\}$!
  - if the decision is yes then perform the y machine
  - if the decision is no then perform the n machine
  - the input can be modified
  - e.g.: $\sqcup aabb\underline{\sqcup} \to y$
    - y is the halting machine

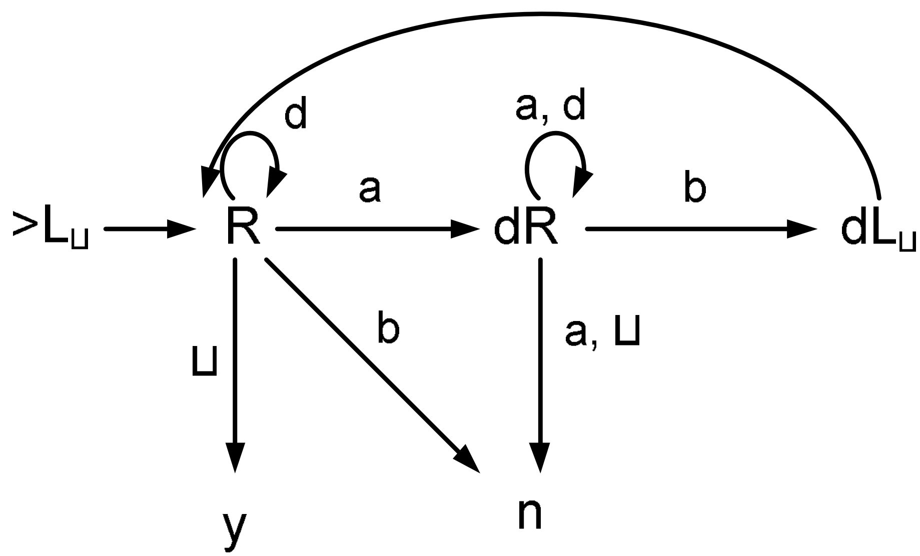
Example 30

- Create a machine schema, which decides
  $L = \{w \in \{a, b\}^* \mid w = a^n b^n\}$!



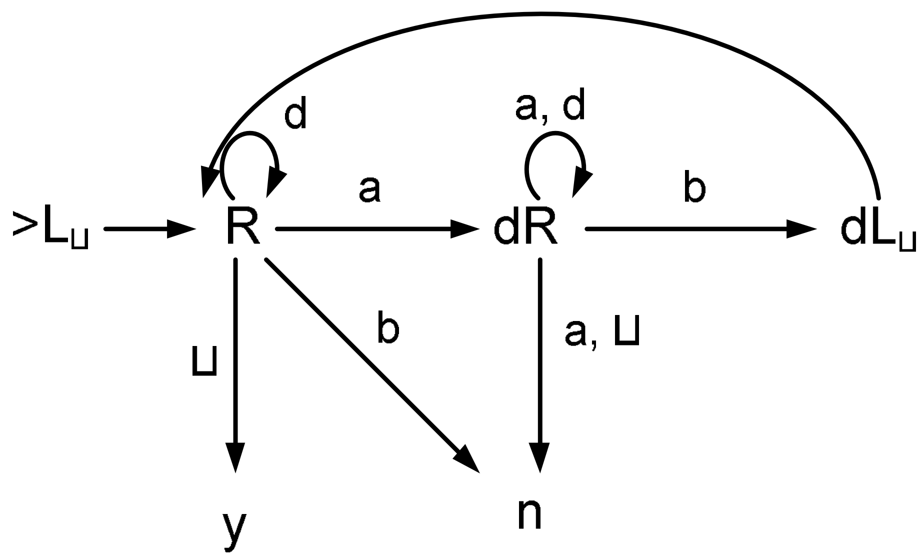
Example 30

- Computation:
  - ⊔aabb$\underline{\sqcup}$
  - $\underline{\sqcup}$aabb
  - ⊔$\underline{a}$abb
  - ⊔$\underline{d}$abb
  - ⊔d$\underline{a}$bb
  - ⊔da$\underline{b}$b
  - ⊔da$\underline{d}$b
  - $\underline{\sqcup}$dadb



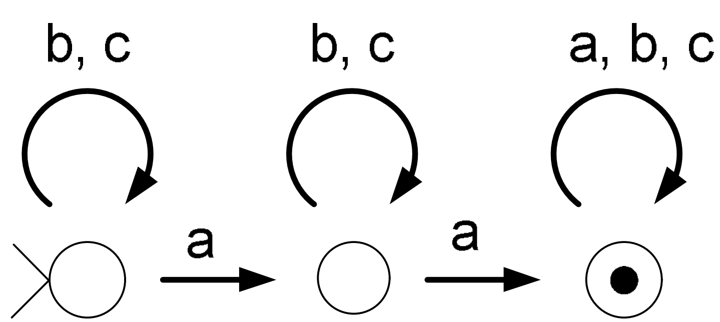
Example 30

- $\sqcup\underline{d}adb$
- $\sqcup d\underline{a}db$
- $\sqcup d\underline{d}db$
- $\sqcup dd\underline{d}b$
- $\sqcup ddd\underline{b}$
- $\sqcup ddd\underline{d}$
- $\underline{\sqcup}dddd$
- ...
- $\sqcup dddd\underline{\sqcup}$ // $\downarrow\sqcup$
- $\sqcup dddd\underline{\sqcup}$

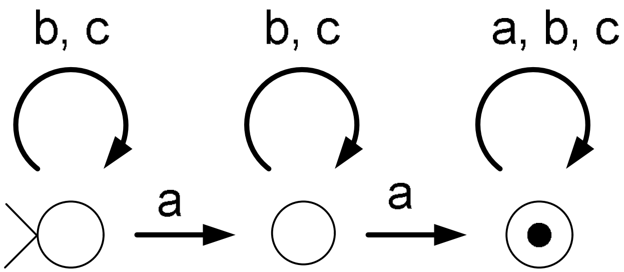


Example 31

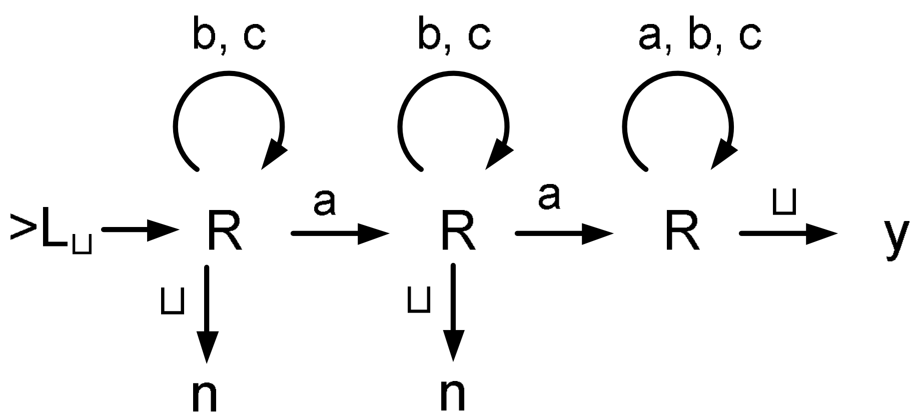
- Create a machine schema, which decides the language of the DFA below!
  - if the decision is yes then perform the y machine
  - if the decision is no then perform the n machine
  - the schema should start from the left side of the input
  - $L = \{w \in \{a, b, c\}^* \mid \#a \geq 2\}$



Example 31



- Create a machine schema, which decides the language of the DFA below!
  - $L = \{w \in \{a, b, c\}^* \mid \#a \geq 2\}$



Example 31

- Computation:
  - $\sqcup abaca\underline{\sqcup}$
  - $\underline{\sqcup}abaca$
  - $\sqcup\underline{a}baca$ // $\rightarrow$a
  - $\sqcup a\underline{b}aca$
  - $\sqcup ab\underline{a}ca$ // $\rightarrow$a
  - $\sqcup aba\underline{c}a$
  - $\sqcup abac\underline{a}$
  - $\sqcup abaca\underline{\sqcup}$ // $\rightarrow\sqcup$
  - $\sqcup abaca\underline{\sqcup}$