Deterministisk endelig automat (DFA) (over språk A)

Består av

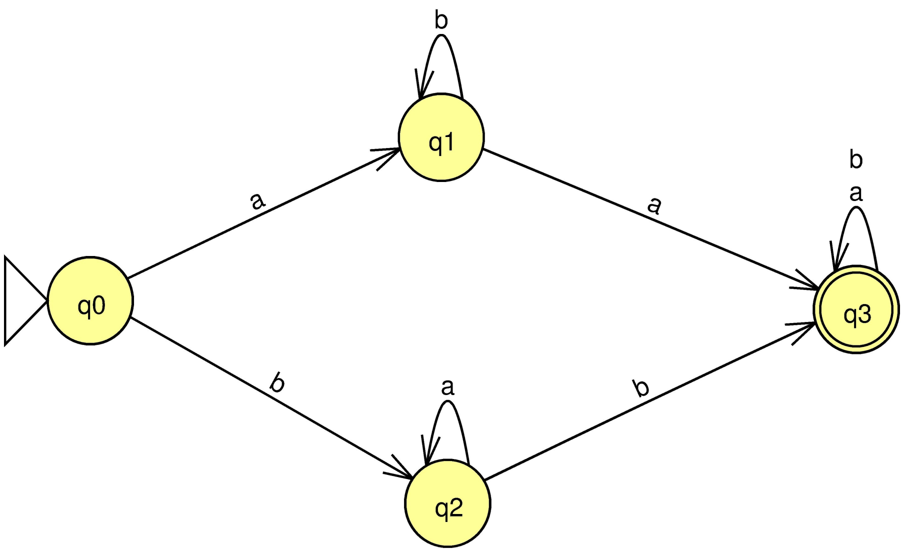
- en ikke-tom mengde Q av *tilstander*

- hvor nøyaktig en er utpekt som *start-tilstand*

- og null eller flere er utpekt som *slutt-tilstander*

- samt en transisjons-funksjon fra $Q \times A$ til Q

Eksempel



A = {a,b}

Q = {q0, q1, q2, q3},

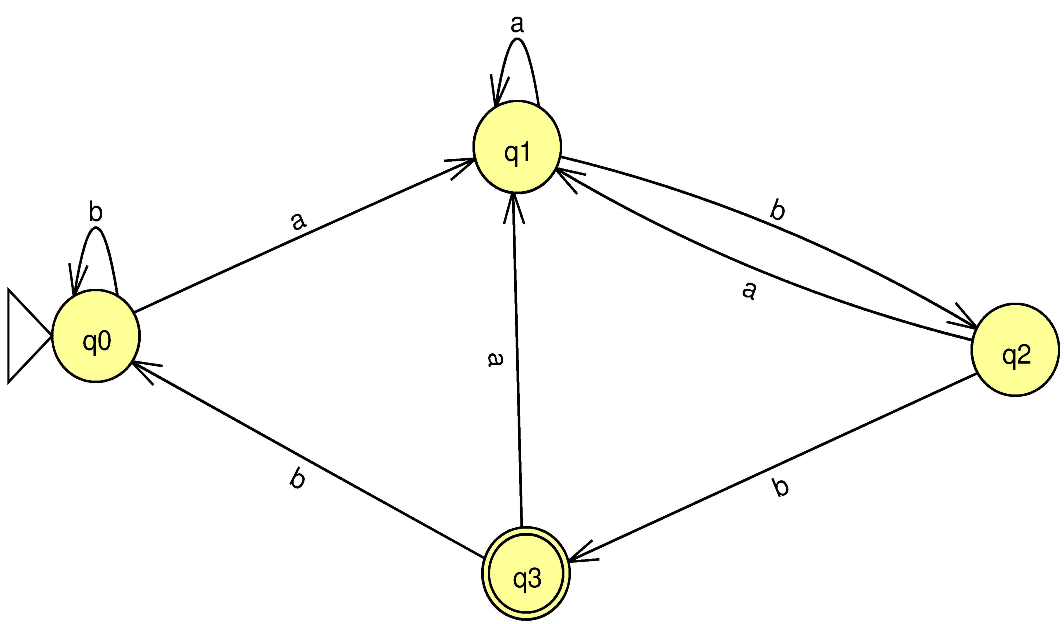
q0 er start-tilstand,

q3 er slutt-tilstand,

transisjons-funksjonen
er gitt ved tabellen

	a	b
q0	q1	q2
q1	q3	q1
q2	q2	q3
q3	q3	q3

Eksempel



A = {a,b}

Q = {q0, q1, q2, q3},

q0 er start-tilstand,

q3 er slutt-tilstand,

transisjons-funksjonen er gitt ved tabellen

	a	b
q0	q1	q0
q1	q1	q2
q2	q1	q3
q3	q1	q0

Automat aksepterer streng:

- En (deterministisk) endelig automat aksepterer en streng hvis vi kan komme fra start-tilstanden til en slutt-tilstand ved å følge strengen.

Automat aksepterer språk:

- En (deterministisk) endelig automat aksepterer et språk hvis den aksepterer alle strenger i språket, og ingen strenger utenfor språket.

Deterministiske endelige automater
vs.
Regulære uttrykk (RE)

- Gir oss det samme!
- Altså: Et språk er regulært hviss det fines en deterministisk endelig automat som aksepterer det.

Bevises oftest ved hjelp av ikke-deterministiske endelige automater

Ikke-deterministisk endelig automat (NFA)