Plan

- Hvad er Regularitet og Automater
- Praktiske oplysninger om kurset
- Regulære udtryk
- Induktionsbevis
- Frokost
- **Endelige automater**
- Skelnelighed, Produktkonstruktion
- Præsentation af Java projekt

Regulære udtryk og endelige automater

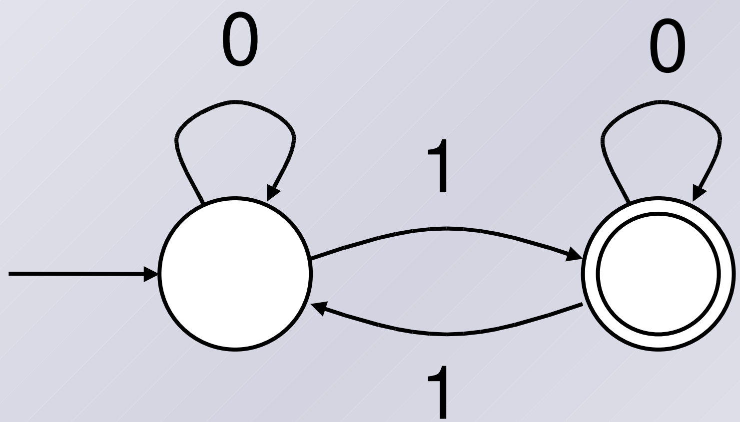
- Regulære udtryk: ***deklarative***
  dvs. ofte velegnede til at *specificere*
  regulære sprog

- Endelige automater: ***operationelle***
  dvs. bedre egnet til at afgøre om
  en given streng er i sproget

- Ethvert regulært udtryk kan oversættes til en endelig automat – og omvendt
  (bevises næste seminar...)

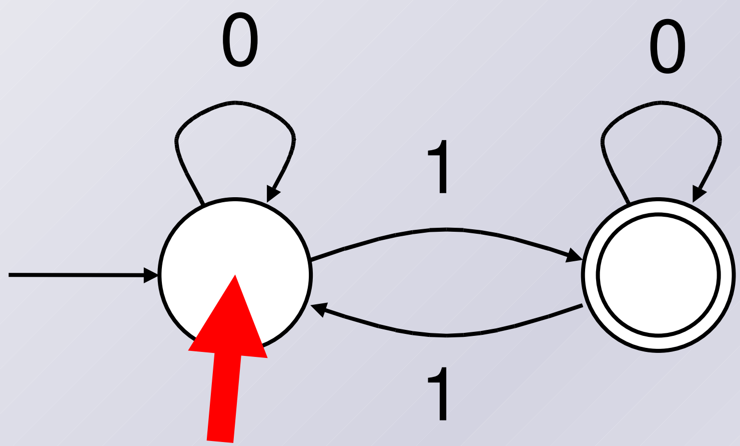
En endelig automat

- En endelig automat, der genkender strenge over alfabetet $\Sigma=\{0,1\}$ med ulige antal 1'er:



- Automaten læser strengen ét tegn ad gangen, fra venstre mod højre
- Hvis automaten ender i en *accept*-tilstand, så accepteres(=genkendes) strengen

At køre en streng på en automat



- Eksempel: vi vil vide om strengen 1010 accepteres
- Vi starter i starttilstanden og læser strengen

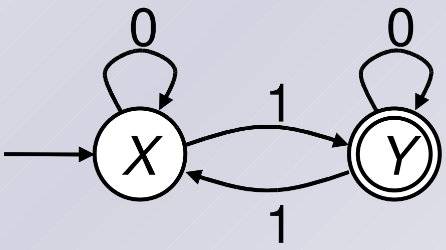
1010

- Vi ender i en ikke-accept tilstand, så strengen accepteres ikke

Hvad repræsenterer tilstandene?

- Hver tilstand repræsenterer en viden om den hidtil læste delstreng

- Eksempel:



- *X*: *"der er læst et **lige** antal 1'er"*
- *Y*: *"der er læst et **ulige** antal 1'er"*

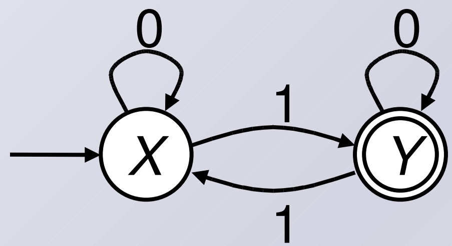
Formel definition af endelige automater

En *endelig automat* (finite automaton/FA) er et 5-tupel $(Q, \Sigma, q_0, A, \delta)$ hvor

- Q er en endelig mængde af tilstande
- Σ er et alfabet
- $q_0 \in Q$ er en starttilstand
- $A \subseteq Q$ er accepttilstande
- $\delta: Q \times \Sigma \rightarrow Q$ er en transitionsfunktion

Eksempel

- Denne grafiske repræsentation af en automat:



- svarer til 5-tuplet (Q, Σ, q_0, A, δ) hvor
  - $Q = \{X, Y\}$

  ∀ $\Sigma = \{0,1\}$
  - $q_0 = X$
  - $A = \{Y\}$

  ∀ δ: $Q \times \Sigma \rightarrow Q$ er denne funktion:

input

tilstand

	0	1
X	*X*	*Y*
Y	*Y*	*X*

Hvorfor en formel definition?

- Den formelle definition viser **kort og præcist** hvad en FA er
- For eksempel,
  - en FA har endeligt mange tilstande
  - den har præcis én starttilstand
  - en vilkårlig delmængde af tilstandene kan være accepttilstande
  - for enhver tilstand q og alfabetsymbol a er der én udgående transition (til tilstanden $\delta(q,a)$)
  - der er ikke noget krav om, at alle tilstande kan nås fra starttilstanden

Sproget af en automat

- 5-tupel-definitionen fortæller hvad en FA *er*
- Vi vil nu definere hvad en FA *kan*:

- En FA ***accepterer*** en streng, hvis dens kørsel fra starttilstanden ender i en accepttilstand
- ***Sproget*** $L(M)$ af en FA M er mængden af strenge, den accepterer
- M ***genkender*** sproget $L(M)$

Formel definition af $L(M)$

- Givet en FA $M=(Q, \Sigma, q_0, A, \delta)$, definer *den udvidede transitionsfunktion* $\delta^*: Q \times \Sigma^* \to Q$ ved

$$\delta^*(q, x) = \begin{cases} q & \text{hvis } x=\Lambda \\ \delta(\delta^*(q, y), a) & \text{hvis } x=ya \text{ hvor } y \in \Sigma^* \text{ og } a \in \Sigma \end{cases}$$

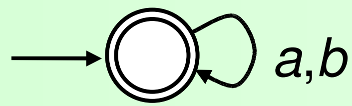
- $x \in \Sigma^*$ *accepteres* af M hvis og kun hvis $\delta^*(q_0, x) \in A$

- Definer $L(M) = \{ x \in \Sigma^* \mid x \text{ accepteres af } M \}$

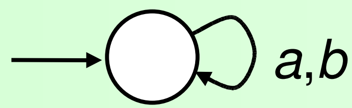
Quiz!

Lad $\Sigma=\{a,b\}$. Konstruér en FA M så...

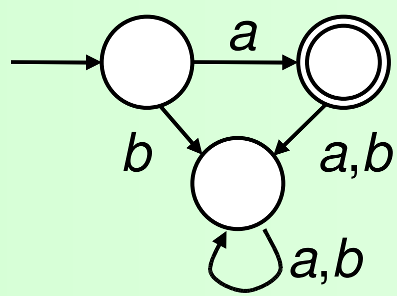
3. $L(M) = \Sigma^*$



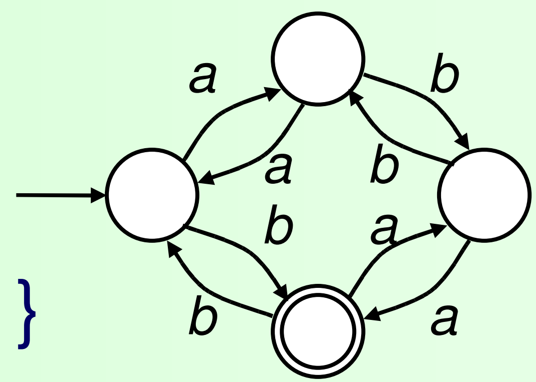
5. $L(M) = \emptyset$



7. $L(M) = \{a\}$



9. $L(M) = \{ x \in \Sigma^* \mid n_a(x)$ lige og $n_b(x)$ ulige $\}$



Øvelser:

- [Martin] Opg. 3.17 (e)
- [Martin] Opg. 3.18
- [Martin] Opg. 3.19 (a-c)