

# Syntaks og semantik

## Lektion 2

8 februar 2007

# Forord

- 1 Administrivia
- 2 Ord
- 3 Sprog
- 4 De regulære operationer
- 5 Regulære udtryk

- Jeg har skrevet en **manual** til kurset; se link på hjemmesiden. Læs den.
- Der var ikke nok *Sipser*-bøger til jer alle. Kopier af kapitel 1 fås ved henvendelse til **Lene**.
- **Syntaksopgaven**

- **alfabet**: en endelig mængde, normalt betegnet  $\Sigma$
- **bogstav / tegn / symbol**: et element i  $\Sigma$
- **ord / streng**: en endelig følge  $(a_1, a_2, \dots, a_k)$  af bogstaver.  
Normalt skrevet uden parenteser og komma:  $a_1 a_2 \dots a_k$
- $\varepsilon$ : det tomme ord (med 0 bogstaver)
- at **sammensætte** ord:  $abe \circ kat = abekat$
- $\varepsilon$  er **identiteten** for  $\circ$ :  $w \circ \varepsilon = \varepsilon \circ w = w$  for alle ord  $w$

- **Sprog (over  $\Sigma$ ):** en mængde af ord med bogstaver fra  $\Sigma$
  - $\emptyset$ : det tomme sprog
  - $\Sigma^*$ : sproget bestående af *alle* ord over  $\Sigma$
- $\Rightarrow L$  er et sprog over  $\Sigma$  hvis og kun hvis  $L \subseteq \Sigma^*$  **quiz (2e)!**

**Bemærk:** Det kan godt være vi snakker om “ord” og “sprog” her, men vi **tillægger dem ikke nogen betydning!** Vi er (lige nu) *kun* interesseret i **formen**, ikke i betydningen.

Givet sprog  $L_1, L_2 \subseteq \Sigma^*$ , da kan vi danne sprogene

- $L_1 \cup L_2 = \{w \mid w \in L_1 \text{ eller } w \in L_2\}$
- $L_1 \circ L_2 = \{w_1 \circ w_2 \mid w_1 \in L_1, w_2 \in L_2\}$
- $L_1^* = \{w_1 \circ w_2 \circ \dots \circ w_k \mid \text{alle } w_i \in L_1\}$

quiz (2c)!

Disse 3 operationer (forening, sammensætning og stjerne) kaldes de **regulære operationer** på sprog.

(Der er andre operationer på sprog, ja.)

quiz (2d)!

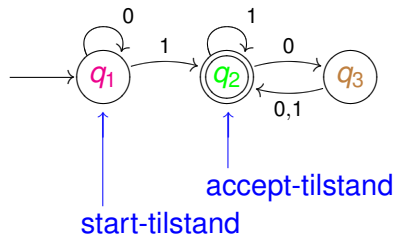
- formål: At beskrive sprog (som generelt er *uendelige* mængder) ved *endelige* udtryk.
- $a$  (for  $a \in \Sigma$ ),  $\varepsilon$ ,  $\emptyset$
- $R_1 \cup R_2$ ,  $R_1 \circ R_2$ ,  $R_1^*$ , for  $R_1, R_2$  regulære udtryk
- en **rekursiv** definition
- forkortelser:  $\Sigma = a_1 \cup a_2 \cup \dots \cup a_n$  (for  $\Sigma = \{a_1, a_2, \dots, a_n\}$ ),  
 $R^+ = R \circ R^*$
- $\llbracket a \rrbracket = \{a\}$ ,  $\llbracket \varepsilon \rrbracket = \{\varepsilon\}$ ,  $\llbracket \emptyset \rrbracket = \emptyset$
- $\llbracket R_1 \cup R_2 \rrbracket = \llbracket R_1 \rrbracket \cup \llbracket R_2 \rrbracket$ ,  $\llbracket R_1 \circ R_2 \rrbracket = \llbracket R_1 \rrbracket \circ \llbracket R_2 \rrbracket$ ,  
 $\llbracket R_1^* \rrbracket = \llbracket R_1 \rrbracket^*$
- ikke alle sprog kan beskrives ved regulære udtryk! (se lektion 4 ...)

# Endelige automater

- 6 Endelige automater
- 7 Eksempler
- 8 Sproget som genkendes af en endelig automat
- 9 At designe endelige automater
- 10 Regulære sprog



- at beskrive sprog ved **maskiner der kan læse dem**
- den mest simple maskine: **endelig automat**
- **tilstande**, og **transitioner** der læser bogstaver:



- eksempel: læs ordet "1101":

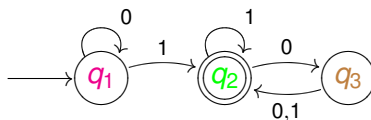
$q_1 \xrightarrow{1} q_2 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \xrightarrow{1} q_2 \Rightarrow \text{accept}$

- eksempel: læs ordet "0110":

$q_1 \xrightarrow{0} q_1 \xrightarrow{1} q_2 \xrightarrow{1} q_2 \xrightarrow{0} q_3 \Rightarrow \text{afvis}$

**Definition 1.5:** En **endelig automat** er en 5-tupel  $(Q, \Sigma, \delta, q_0, F)$ , hvor delene er

- 1  $Q$  : en endelig mængde af **tilstande**
- 2  $\Sigma$  : en endelig mængde af **bogstaver** (input-alfabetet)
- 3  $\delta : Q \times \Sigma \rightarrow Q$  : **transitions-funktionen**
- 4  $q_0 \in Q$  : **starttilstanden**
- 5  $F \subseteq Q$  : mængden af **accepttilstande**



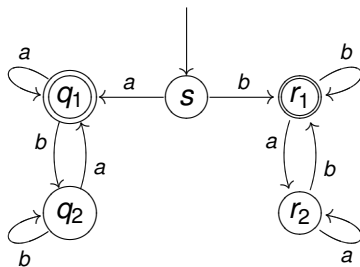
Her har vi:

- 1 tilstande  $Q = \{q_1, q_2, q_3\}$
- 2 inputalfabetet  $\Sigma = \{0, 1\}$
- 3 transitionsfunktionen  $\delta : Q \times \Sigma \rightarrow Q$  givet ved

	0	1
$q_1$	$q_1$	$q_2$
$q_2$	$q_3$	$q_2$
$q_3$	$q_2$	$q_2$

- 4 starttilstanden  $q_0 = q_1$
- 5 accepttilstandene  $F = \{q_2\}$

## Eksempel 1.11:



$$Q = \{s, q_1, q_2, r_1, r_2\}$$

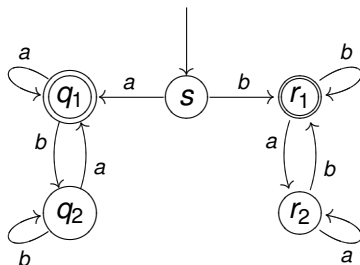
$$\Sigma = \{a, b\}$$

$$q_0 = s$$

$$F = \{q_1, r_1\}$$

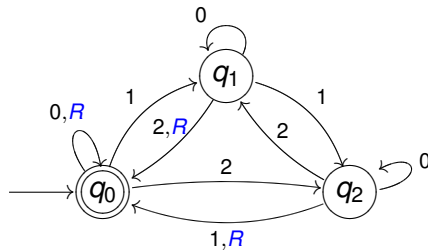
$\delta :$	$a$	$b$
$s$	$q_1$	$r_1$
$q_1$	$q_1$	$q_2$
$q_2$	$q_1$	$q_2$
$r_1$	$r_2$	$r_1$
$r_2$	$r_2$	$r_1$

## Eksempel 1.11:



Accepterer alle ord der starter og slutter med samme bogstav.

### Eksempel 1.13:



Accepterer alle ord hvor summen af cifrene efter det sidste “*R*” er deleligt med 3 !

**Eksempel 1.15:** En endelig automat over alfabetet  $\{0, 1, 2, R\}$  der accepterer alle ord hvor summen af cifrene efter det sidste " $R$ " er deleligt med et givet tal  $i$ :

$$Q = \{q_0, q_1, \dots, q_{i-1}\}$$

$$\Sigma = \{0, 1, 2, R\}$$

$$q_0 = q_0$$

$$F = \{q_0\}$$

$$\delta(q_j, 0) = q_j$$

$$\delta(q_j, 1) = q_{j+1 \bmod i}$$

$$\delta(q_j, 2) = q_{j+2 \bmod i}$$

$$\delta(q_j, R) = q_0$$

– kan umiddelbart generaliseres til

$$\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, R\} \quad (\text{Hvordan?})$$

**Definition:** Lad  $M = (Q, \Sigma, \delta, q_0, F)$  være en endelig automat, og lad  $w = w_1 w_2 \dots w_n \in \Sigma^*$ . Da siges  $M$  at **acceptere**  $w$  hvis der findes en følge  $(r_0, r_1, \dots, r_n)$  af tilstande  $r_i \in Q$  således at

- 1  $r_0 = q_0$ ,
- 2  $r_{i+1} = \delta(r_i, w_{i+1})$  for alle  $i = 0, 1, \dots, n-1$ , og
- 3  $r_n \in F$ .

Sproget som **genkendes** af  $M$  er

$$\llbracket M \rrbracket = L(M) = \{w \mid M \text{ accepterer } w\}$$



**Eksempel: Sætning:** Sproget som genkendes af automaten  $M$  fra eksempel 1.15 er

$L = \{w \mid \text{summen af cifrene efter sidste "R" er deleligt med } i\}$

**Bevis:** Lad  $w \in \Sigma^*$ , og skriv  $w$  som  $w = \Sigma^* R w_1 w_2 \dots w_k$ , hvor  $w_1, w_2, \dots, w_k \in \{0, 1, 2\}$ . Dvs.  $w_1 w_2 \dots w_k$  er den del af  $w$  der står efter det sidste "R."

Efter at have læst det sidste "R," er  $M$  i tilstand  $q_0$ . Lad nu  $r_1, r_2, \dots, r_k$  betegne de tilstande som  $M$  er i efter at have læst  $w_1, w_2, \dots, w_k$ . Da er

$$r_1 = \delta(q_0, w_1) = q_{w_1 \bmod i}$$

$$r_2 = \delta(r_1, w_2) = \delta(q_{w_1 \bmod i}, w_2) = q_{w_1 + w_2 \bmod i}$$

$$r_3 = \delta(r_2, w_3) = \delta(q_{w_1 + w_2 \bmod i}, w_3) = q_{w_1 + w_2 + w_3 \bmod i}$$

$$\vdots$$

$$r_k = q_{w_1 + w_2 + \dots + w_k \bmod i}$$

Bemærk nu at  $w$  accepteres af  $M$  hvis og kun hvis  $r_k = q_0$ .  
Dvs.  $w$  accepteres af  $M$  hvis og kun hvis

$$w_1 + w_2 + \dots + w_k \bmod i = 0. \quad \square$$

Clue: tilstandene repræsenterer *information*!

**Eksempel 1.21:** En endelig automat der genkender sproget  $\Sigma^*001\Sigma^*$ , for  $\Sigma = \{0, 1\}$ .

- starttilstand
- tilstand “jeg har lige set ‘0’ ”
- tilstand “jeg har lige set ‘00’ ”
- tilstand “jeg har lige set ‘001’ ” (accept!)

Clue: tilstandene repræsenterer *information*!

**Eksempel 1.21:** En endelig automat der genkender sproget  $\Sigma^*001\Sigma^*$ , for  $\Sigma = \{0, 1\}$ .

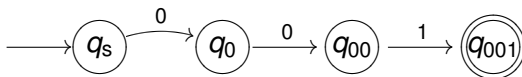
- starttilstand  $q_s$
- tilstand "jeg har lige set '0' "  $q_0$
- tilstand "jeg har lige set '00' "  $q_{00}$
- tilstand "jeg har lige set '001' " (accept!)  $q_{001}$



Clue: tilstandene repræsenterer *information*!

**Eksempel 1.21:** En endelig automat der genkender sproget  $\Sigma^*001\Sigma^*$ , for  $\Sigma = \{0, 1\}$ .

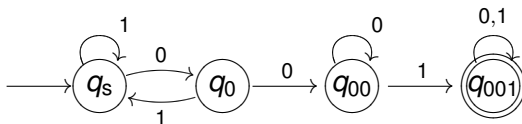
- starttilstand  $q_s$
- tilstand "jeg har lige set '0' "  $q_0$
- tilstand "jeg har lige set '00' "  $q_{00}$
- tilstand "jeg har lige set '001' " (accept!)  $q_{001}$



Clue: tilstandene repræsenterer *information*!

**Eksempel 1.21:** En endelig automat der genkender sproget  $\Sigma^*001\Sigma^*$ , for  $\Sigma = \{0, 1\}$ .

- starttilstand  $q_s$
- tilstand "jeg har lige set '0' "  $q_0$
- tilstand "jeg har lige set '00' "  $q_{00}$
- tilstand "jeg har lige set '001' " (accept!)  $q_{001}$



**Definition 1.16:** Et sprog siges at være **regulært** hvis der findes en endelig automat der genkender det.

*Eller:* Givet et alfabet  $\Sigma$  og  $L \subseteq \Sigma^*$ , da kaldes  $L$  et **regulært sprog** hvis der findes en endelig automat  $M$  over  $\Sigma$  således at  $\llbracket M \rrbracket = L$ .

**Vigtig sætning 1.54:** Et sprog er regulært hvis og kun hvis det kan beskrives ved et **regulært udtryk**.

(Beviset ser vi på næste gang.)

### Sætning 1.25 / 1.45 / 1.47 / 1.49:

Klassen af regulære sprog er **lukket** under foreningsmængde  $\cup$ , sammensætning  $\circ$  og stjerne  $^*$ .

Dvs. hvis  $A$  og  $B$  er regulære sprog, da er også

- $A \cup B$ ,
- $A \circ B$  og
- $A^*$

regulære sprog.

Beviserne skal vi se i dag og næste gang.

**Sætning 1.25:** Lad  $A_1$  og  $A_2$  være regulære sprog over et fælles alfabet  $\Sigma$ . Da er også  $A_1 \cup A_2$  et regulært sprog.

**Bevis:** Lad  $M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1)$ ,  $M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2)$  være endelige automater med  $\llbracket M_1 \rrbracket = A_1$  og  $\llbracket M_2 \rrbracket = A_2$ . Konstruér en ny endelig automat  $M = (Q, \Sigma, \delta, q_0, F)$  ved

- $Q = Q_1 \times Q_2$ ,
- $q_0 = (q_1, q_2)$ ,
- $F = \{(r_1, r_2) \in Q \mid r_1 \in F_1 \text{ eller } r_2 \in F_2\}$ ,
- og med  $\delta : Q \times \Sigma \rightarrow Q$  defineret som

$$\delta((r_1, r_2), a) = (\delta(r_1, a), \delta(r_2, a))$$

For at vise at  $\llbracket M \rrbracket = A_1 \cup A_2$ , skal vi vise at

- 1 ethvert  $w \in \llbracket M_1 \rrbracket$  også er i  $\llbracket M \rrbracket$ ,
- 2 ethvert  $w \in \llbracket M_2 \rrbracket$  også er i  $\llbracket M \rrbracket$ , og at
- 3 ethvert  $w \in \llbracket M \rrbracket$  også er i  $\llbracket M_1 \rrbracket$  eller i  $\llbracket M_2 \rrbracket$ .