

Syntaksopgaven

Et tip / ønske til syntaksopgaven:

Indfør 4 alfabeter:

$$\begin{aligned}\Sigma_0 &= \{0\} \\ \Sigma_1 &= \{1, 2, \dots, 9\} \\ \Sigma_2 &= \{a, \dots, z, A, \dots, Z\} \\ \Sigma_3 &= \{+, -, *\}\end{aligned}$$

Sæt $\Sigma = \Sigma_0 \cup \Sigma_1 \cup \Sigma_2 \cup \Sigma_3$, og betragt alle automater og udtryk over alfabetet Σ .

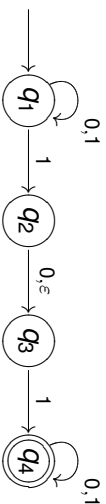
Brug $\Sigma_0, \Sigma_1, \Sigma_2, \Sigma_3$ som forkortelser på automaters pile og i udtrykkene.

Forord

- 1 Administrivia
- 2 Non-deterministiske endelige automater
- 3 NFAs og regulære udtryk
- 4 Eksempel på delmængdekonstruktion
- 5 Transitionssystemer

Planen

- i dag: afslutning på kursusdelen omhandlende regulære sprog
- og afslutning på syntaksopgavens del omhandlende regulære sprog
- næste gang: perspektivering og spørgetime !
- og start på kontekstfrie sprog



Definition 1.37: En **nondeterministisk endelig automat (NFA)** er en 5-tupel $M = (Q, \Sigma, \delta, q_0, F)$, hvor delene er

- 1 Q : en endelig mængde af tilstande
- 2 Σ : input-alfabetet
- 3 $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \rightarrow \mathcal{P}(Q)$: transitions-funktionen
- 4 $q_0 \in Q$: starttilstanden
- 5 $F \subseteq Q$: mængden af accepttilstande

M siges at **acceptere** et ord $w \in \Sigma^*$ hvis der findes $m \in \mathbb{N}$ og $y_1, y_2, \dots, y_m \in \Sigma \cup \{\varepsilon\}$ og $r_0, r_1, \dots, r_m \in Q$ således at $w = y_1 y_2 \dots y_m$ og

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

5/24

- enhver DFA er også en NFA
- enhver NFA kan laves om til en DFA der genkender samme sprog (**delmængdekonstruktionen**)
- et sprog er defineret til at være **regulært** hvis der er en DFA der genkender det
- \Rightarrow et sprog er regulært hvis og kun hvis der er en NFA der genkender det
- regulære sprog er lukket under $\cup, \circ, *$ (vises ved at konstruere en ny NFA ud fra de givne NFAs)
- regulære sprog er lukket under \cap og $-$ (komplement) (vises ved at konstruere en ny DFA ud fra de givne DFAs; konstruktionerne virker *kun* for DFAs!)
- NFAs er generelt mere simple at fremstille
- NFA = **abstraktion** !

6/24

Lemma 1.55: Hvis et sprog beskrives ved et regulært udtryk, da er det regulært.

Bevises ved strukturel induktion:

- 1 konvertér de basale regulære udtryk til NFAs
- 2 brug lukningsegenskaber til at konvertere sammensætninger af regulære udtryk til sammensætninger af NFAs
- 3 **Smart!**

I dag: **Lemma 1.60:** Hvis et sprog er regulært, da kan det beskrives ved et regulært udtryk.

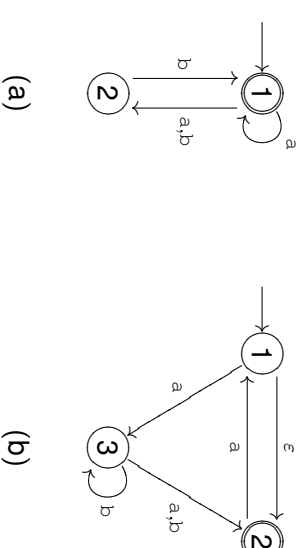
(Bevises ved at *generalisere* NFAs til **GNFAs**.)

\Rightarrow **Sætning 1.54:** Et sprog er regulært **hvis og kun hvis** det kan beskrives ved et regulært udtryk.

7/24

Opgave 1.16

Konvertér følgende to NFAs til DFAs ved hjælp af delmængdekonstruktionen: (**ved tavlen**)



8/24

Transitions-systemer: en generalisering af endelige automater, både DFA og NFA.

Definition: Et **transitionssystem** er en 4-tupel (Q, Σ, T, q_0) , hvor delene er

- 1 Q : en mængde af tilstande (**endelig eller uendelig**)
- 2 Σ : et alfabet (**en endelig mængde**)
- 3 $T \subseteq Q \times \Sigma \times Q$: **transitions-relationen**
- 4 $q_0 \in Q$: starttilstanden

- En NFA er et **endeligt** transitionssystem med en specificeret mængde af **accepttilstande**, og med et specielt tegn $\varepsilon \in \Sigma$
- En DFA er en NFA som opfylder følgende egenskaber:
 - 1 der er ingen transitioner $(q, \varepsilon, q') \in T$
 - 2 for alle $q \in Q$ og alle $a \in \Sigma$, med $a \neq \varepsilon$, findes $q' \in Q$ og en transition $(q, a, q') \in T$
 - 3 hvis $(q, a, q'_1) \in T$ og $(q, a, q'_2) \in T$, så er $q'_1 = q'_2$

9 / 24

NFA \Rightarrow RE

Ikke-regulære sprog

Regulære og ikke-regulære sprog

- 6 Regulære sprog genereres af regulære udtryk
- 7 Ikke-regulære sprog

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

Nøgle til beviset: Ny slags maskiner der kombinerer NFA og regulære udtryk: **generaliserede nondeterministiske endelige automater (GNFA)**

Definition 1.64: En **GNFA** er en 5-tupel $(Q, \Sigma, \delta, q_0, q_f)$, hvor delene er

- 1 Q : en endelig mængde af tilstande
- 2 Σ : input-alfabetet
- 3 $\delta : (Q \setminus \{q_f\}) \times (Q \setminus \{q_0\}) \rightarrow \mathcal{R}$: transitions-funktionen
- 4 $q_0 \in Q$: starttilstanden
- 5 $q_f \in Q$: accepttilstanden

Notation: $\mathcal{R} = \mathcal{R}(\Sigma) =$ mængden af alle **regulære udtryk** over et givet alfabet Σ .

(Bemærk at GNFA's introduceres kun for det her bevis. De bruges ikke til andet.)

11 / 24

NFA \Rightarrow RE

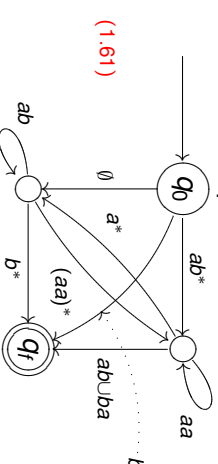
Ikke-regulære sprog

Definition 1.64: En **GNFA** er en 5-tupel $(Q, \Sigma, \delta, q_0, q_f)$, hvor delene er

- 3 $\delta : (Q \setminus \{q_f\}) \times (Q \setminus \{q_0\}) \rightarrow \mathcal{R}$: transitions-funktionen
- 5 $q_f \in Q$: accepttilstanden

Ligesom NFAs, men

- med kun **én** accepttilstand
- med **regulære udtryk** på transitionerne i stedet for tegn
- med transitioner **fra enhver tilstand til enhver tilstand** (også sig selv), bortset fra at
 - starttilstanden ikke har **indgående** transitioner, og at
 - accepttilstanden ikke har **udgående** transitioner



Endnu en speciel form for **transitionssystem**: **alfabetet** er $\mathcal{R}(\Sigma)$, så transitionerne er $T \subseteq Q \times \mathcal{R}(\Sigma) \times Q$.

Definition 1.64: En **GNFA** er en 5-tupel $(Q, \Sigma, \delta, q_0, q_f)$, hvor delene er

- 1 Q : en endelig mængde af tilstande
- 2 Σ : input-alfabetet
- 3 $\delta : (Q \setminus \{q_f\}) \times (Q \setminus \{q_0\}) \rightarrow \mathcal{R}$: transitions-funktionen
- 4 $q_0 \in Q$: starttilstanden
- 5 $q_f \in Q$: accepttilstanden

GNFAen **accepterer** et ord $w \in \Sigma^*$ hvis der findes $m \in \mathbb{N}$ og $y_1, y_2, \dots, y_m \in \Sigma^*$ (!) og $r_0, r_1, \dots, r_m \in Q$ således at $w = y_1 y_2 \dots y_m$ og

- 1 $r_0 = q_0$,
- 2 $y_{i+1} \in \llbracket \delta(r_i, r_{i+1}) \rrbracket$ for alle $i = 0, 1, \dots, m-1$, og
- 3 $r_m = q_f$.

Bevisidé: konvertér en DFA til en GNFA og så GNFAen til et regulært udtryk ved at **fjerne én tilstand ad gangen**.

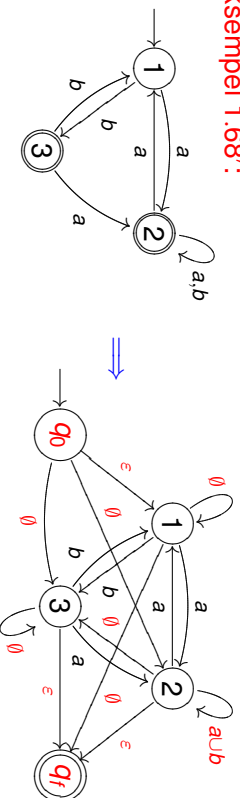
13/24

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

Bewis: Lad $M = (Q, \Sigma, \delta, q_1, F_1)$ være en DFA med $\llbracket M \rrbracket = L$.

- 1 Konvertér M til en GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$:
 - (a) Lav en ny starttilstand q_0 og en ny accepttilstand q_f , med ε -transitioner fra q_0 til den gamle starttilstand og fra alle gamle accepttilstande til q_f .
 - (b) Erstat transitioner med flere end ét label med én transition der som label har foreningen af disse labels.
 - (c) Indsæt \emptyset -transitioner hvor der mangler pile.

Eksempel 1.68':



14/24

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

Bewis: Lad $M = (Q, \Sigma, \delta, q_1, F_1)$ være en DFA med $\llbracket M \rrbracket = L$.

- 1 Konvertér M til en GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$:
 - (a) Lav en ny starttilstand q_0 og en ny accepttilstand q_f , med ε -transitioner fra q_0 til den gamle starttilstand og fra alle gamle accepttilstande til q_f .
 - (b) Erstat transitioner med flere end ét label med én transition der som label har foreningen af disse labels.
 - (c) Indsæt \emptyset -transitioner hvor der mangler pile.

$$\delta(q, q') = \begin{cases} \varepsilon & \text{hvis } q = q_0 \text{ eller } q' = q_f \\ a_1 \cup a_2 \cup \dots \cup a_k & \text{hvis } q, q' \in Q_1 \text{ og } \delta_1(q, a_i) = q' \\ \emptyset & \text{for alle } i = 1, 2, \dots, k \\ & \text{hvis } q, q' \in Q_1 \text{ og } \delta_1(q, a) \neq q' \\ & \text{for alle } a \in \Sigma \end{cases}$$

15/24

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

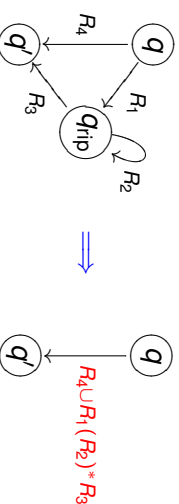
Bewis: Lad $M = (Q, \Sigma, \delta, q_1, F_1)$ være en DFA med $\llbracket M \rrbracket = L$.

- 1 Konvertér M til en GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$
- 2 Konvertér G til et regulært udtryk R :

CONVERT(G):

- 1 Lad $k = |Q|$ – antallet af tilstande i G .
- 2 Hvis $k = 2$, returnér $\delta(q_0, q_f)$.
- 3 Vi har $k > 2$. Lad $q_{rip} \in Q \setminus \{q_0, q_f\}$.

Lad $Q' = Q \setminus \{q_{rip}\}$, og definér $\delta' : (Q' \setminus \{q_f\}) \times (Q' \setminus \{q_0\}) \rightarrow \mathcal{R}$ på følgende måde:



16/24

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

Bevis: Lad $M = (Q, \Sigma, \delta, q_1, F_1)$ være en DFA med $\llbracket M \rrbracket = L$.

- 1 Konvertér M til en GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$
- 2 Konvertér G til et regulært udtryk R :

CONVERT(G):

- 1 Lad $k = |Q|$ – antallet af tilstande i G .
 - 2 Hvis $k = 2$, returnér $\delta(q_0, q_f)$.
 - 3 Vi har $k > 2$. Lad $q_{rip} \in Q \setminus \{q_0, q_f\}$.
- Lad $Q' = Q \setminus \{q_{rip}\}$, og definér
- $$\delta' : (Q' \setminus \{q_f\}) \times (Q' \setminus \{q_0\}) \rightarrow \mathcal{R} \text{ på følgende måde:}$$

For $q \in Q' \setminus \{q_f\}$ og $q' \in Q' \setminus \{q_0\}$ lad
 $R_1 = \delta(q, q_{rip})$, $R_2 = \delta(q_{rip}, q_{rip})$, $R_3 = \delta(q_{rip}, q')$
og $R_4 = \delta(q, q')$, og lad
 $\delta'(q, q') = R_4 \cup R_1(R_2)^*R_3$.

- 4 Returnér **CONVERT**($G' = (Q', \Sigma, \delta', q_0, q_f)$)

17/24

Lemma 1.60: Givet et alfabet Σ og et regulært sprog $L \subseteq \Sigma^*$, da findes et regulært udtryk R over Σ således at $L = \llbracket R \rrbracket$.

Bevis: Lad $M = (Q, \Sigma, \delta, q_1, F_1)$ være en DFA med $\llbracket M \rrbracket = L$.

- 1 Konvertér M til en GNFA $G = (Q, \Sigma, \delta, q_0, q_f)$
- 2 Konvertér G til et regulært udtryk R .
- 3 Vis at $\llbracket M \rrbracket = \llbracket R \rrbracket$:
 - 1 Vis at $\llbracket M \rrbracket = \llbracket G \rrbracket$: nemt
 - 2 Vis at $\llbracket G \rrbracket = \llbracket R \rrbracket$:
 - 1 Hvis $k = |Q| = 2$: $Q = \{q_0, q_f\}$, og $R = \delta(q_0, q_f) \Rightarrow \checkmark$
 - 2 Hvis $k > 2$: Vis at $\llbracket G \rrbracket = \llbracket G' \rrbracket$
- 4 Done!

18/24

Ikke alle sprog er regulære. F.x. sproget $\{0^n 1^n \mid n \in \mathbb{N}\}$:



– en **uendelig** automati!

Pumping Lemma: en egenskab ved alle regulære sprog.

\Rightarrow Hvis et sprog **ikke** har den egenskab, **kan det ikke være regulært**.

19/24

Sætning 1.70 (Pumpelæmmaet): For ethvert regulært sprog A findes der et (naturligt) tal p således at ethvert ord $s \in A$ der har længde mindst p kan opsplittes i tre stykker, $s = xyz$, med

- $|y| > 0$ og $|xy| \leq p$,
- og således at ordene $xy^i z \in A$ for alle $i \in \mathbb{N}_0$.

En gang til:

For ethvert regulært sprog A

findes $p \in \mathbb{N}_0$ således at

for ethvert $s \in A$ med $|s| \geq p$

findes en opsplitning $s = xyz$ således at

$|y| > 0$ og $|xy| \leq p$ og
for alle $i \in \mathbb{N}_0$
 $xy^i z \in A$.

20/24

Sætning 1.70 (Pumpelemmaet): For ethvert regulært sprog A findes der et (naturligt) tal p således at ethvert ord $s \in A$ der har længde mindst p kan opsplittes i tre stykker, $s = xyz$, med

- $|Y| > 0$ og $|XY| \leq p$,
- og således at ordene $xy^i z \in A$ for alle $i \in \mathbb{N}_0$.

Eksempel 1.7.3: Sproget $B = \{0^n 1^n \mid n \in \mathbb{N}\}$ er ikke regulært.

Bevis (ved modstrid; kortere end i bogen!): Antag at B er reguleret, og lad p være pumpe længden. Lad $s = 0^p 1^p$, da er $|s| \geq p$.

Lad $s = xyz$ være en opsplitning af s som opfylder pumpelemmets betingelser. Pga. $|xy| \leq p$ kan y kun indeholde 0'er, og pga. $|y| > 0$ indeholder y mindst ét 0.

Sidste betingelse i lemmaet siger bl.a. at orden $xyz \in A$, men dette ord indeholder for mange 0'er. **Modstrid!**

Sætning 1.70 (Pumpelemmaet): For ethvert reguleret sprog A findes der et (naturligt) tal p således at ethvert ord $s \in A$ der har længde mindst p kan opsplittes i tre stykker, $s = xyz$, med

- $|Y| > 0$ og $|XY| \leq p$,
- og således at ordene $xy^i z \in A$ for alle $i \in \mathbb{N}_0$.

Bevis: Lad $M = (Q, \Sigma, \delta, q_0, F)$ være en DFA der genkender A , og lad $p = |Q|$. Lad $s = s_1 s_2 \dots s_n \in A$ med $|s| \geq p$.

Mens M læser s , kommer den igennem en følge af $n + 1$ tilstande.

Men $n+1 > p$, så der er flere tilstande i følgen end der er i M .

Dvs. der er en tilstand der optræder to gange i følgen – en løkke!

Hvis vi tager x til at være den del af s der læses *før* løkken, y den del der læses *i* løkken, og z den del der læses *efter* løkken, kan vi gennemløbe løkken i gange og genkende strengen xy^iz .

Sætning 1.70 (Pumpelemmaet): For ethvert reguleret sprog A findes der et (naturligt) tal p således at ethvert ord $s \in A$ der har længde mindst p kan opsplittes i tre stykker, $s = xyz$, med

- $|y| > 0$ og $|xy| \leq p$,
- og således at ordene $xy^i z \in A$ for alle $i \in \mathbb{N}_0$.

Bævis: Lad $M = (Q, \Sigma, \delta, q_0, F)$ være en DFA der genkender A , og lad $p = |Q|$. Lad $s = s_1s_2 \dots s_n \in A$ med $|s| \geq p$.

Lad $r_1, r_2, \dots, r_{n+1} \in Q$ således at $r_1 = q_0, r_{n+1} \in F$, og $r_{i+1} = \delta(r_i, s_i)$ for alle i .

Vi har $n+1 \geq p+1$, $\text{og } |Q| = p$. Derfor findes indices j og ℓ således at $1 \leq j < \ell \leq p+1$ og $r_j = r_\ell$.

Lad $x = s_1 \dots s_{j-1}$, $y = s_j \dots s_{\ell-1}$, $z = s_{\ell} \dots s_n$. Pga. $j < \ell$ har vi $|y| \geq 0$, og $\ell \leq p + 1$ medfører $|xy| \leq p$.

Eftersom $\delta(r_{\ell-1}, s_{\ell-1}) = f_j$, er enhver følge $(r_1, \dots, f_{j-1})(r_j, \dots, r_{\ell-1})(r_{\ell}, \dots, r_{n+1})$ en accepterende følge for M , og ordet den genkender er $xy^{\ell}z$.

Eksempel 1.7.4: Sproget

$C = \{w \mid \text{antallet af } 0 \text{ i } w \text{ er lig med antallet af } 1\} \subseteq \{0, 1\}^*$ er ikke regulært.

(Samme bevis som for eksempel 1.73)

Bemærkning (opgave 1.48): Sproget

$D = \{w \mid \text{antallet af 0' i } w \text{ er lig med antallet af 1'0}\} \subseteq \{0,1\}^*$ er regulært!

(Men kun over alfabetet $\{0, 1\}$; hvis alfabetet f.x. er $\{0, 1, 2\}$, er D ikke regulært ...)

Bevis:

