

Syntaks og semantik

Lektion 6

26 februar 2008

Fra sidst

- 1 Kontekstfrie grammatikker
- 2 Lukningsegenskaber
- 3 Regulære grammatikker

Definition 2.2: En **kontekstfri grammatik (CFG)** er en 4-tupel $G = (V, \Sigma, R, S)$, hvor delene er

- ① V : en endelig mængde af **variable**
- ② Σ : en endelig mængde af **terminaler**, med $V \cap \Sigma = \emptyset$
- ③ $R : V \rightarrow \mathcal{P}((V \cup \Sigma)^*)$: **produktioner / regler**
- ④ $S \in V$: **startvariablen**

– produktioner skrives $A \rightarrow w$ i stedet for $w \in R(A)$

- Hvis $u, v, w \in (V \cup \Sigma)^*$ er ord og $A \rightarrow w$ er en produktion, siges uAv at **frembringe** uwv : $uAv \Rightarrow uwv$.
- Hvis $u, v \in (V \cup \Sigma)^*$ er ord, siges u at **derivere** v : $u \xRightarrow{*} v$, hvis $u = v$ (!) eller der findes en følge u_1, u_2, \dots, u_k af ord således at $u \Rightarrow u_1 \Rightarrow u_2 \Rightarrow \dots \Rightarrow u_k \Rightarrow v$.
- **Sproget** som G genererer er $\llbracket G \rrbracket = \{w \in \Sigma^* \mid S \xRightarrow{*} w\}$.

– dvs. et ord $w \in \Sigma^*$ genereres af G hvis og kun hvis der findes en **derivation** $S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \dots \Rightarrow w_k \Rightarrow w$, hvor alle $w_i \in (V \cup \Sigma)^*$.

Eksempel: Opgave 2.6 d (ca.)

$$S \rightarrow A\#T\#A$$

$$T \rightarrow aTa \mid bTb \mid \#A\#$$

$$A \rightarrow aA \mid bA \mid \varepsilon \mid A\#A$$

Genererer sproget

$$\{x_1\#x_2\#\dots\#x_k \mid k \geq 5, \text{ alle } x_i \in \{a, b\}^*,$$

$$\text{og } x_i = x_j^R \text{ for to indices } i \neq j\}$$

Definition: Et sprog siges at være **kontekstfrit** hvis der findes en CFG der genererer det.

Sætning 2.20: Et sprog er kontekstfrit hvis og kun hvis der findes en **push-down-automat** der genkender det.

(PDAs kommer lige om lidt.)

Sætning: Klassen af kontekstfrie sprog er lukket under \cup , \circ og $*$.

Bevis: (Opgave 2.8) Lad A_1 og A_2 være kontekstfrie sprog over et fælles alfabet Σ .

- \cup : Lad $G_1 = (V_1, \Sigma, R_1, S_1)$, $G_2 = (V_2, \Sigma, R_2, S_2)$ være CFGs med $\llbracket G_1 \rrbracket = A_1$ og $\llbracket G_2 \rrbracket = A_2$. Konstruér en ny CFG $G = (V, \Sigma, R, S)$ ved $V = V_1 \cup V_2 \cup \{S\}$ og $R = R_1 \cup R_2 \cup \{S \rightarrow S_1 \mid S_2\}$. Da er $\llbracket G \rrbracket = A_1 \cup A_2$.
- \circ : Lad $G_1 = (V_1, \Sigma, R_1, S_1)$, $G_2 = (V_2, \Sigma, R_2, S_2)$ være CFGs med $\llbracket G_1 \rrbracket = A_1$ og $\llbracket G_2 \rrbracket = A_2$. Konstruér en ny CFG $G = (V, \Sigma, R, S)$ ved $V = V_1 \cup V_2 \cup \{S\}$ og $R = R_1 \cup R_2 \cup \{S \rightarrow S_1 S_2\}$. Da er $\llbracket G \rrbracket = A_1 \circ A_2$.
- $*$: Lad $G_1 = (V_1, \Sigma, R_1, S_1)$ være en CFG med $\llbracket G_1 \rrbracket = A_1$. Konstruér en ny CFG $G = (V, \Sigma, R, S)$ ved $V = V_1 \cup \{S\}$ og $R = R_1 \cup \{S \rightarrow \varepsilon \mid SS \mid S_1\}$. Da er $\llbracket G \rrbracket = A_1^*$.

- **Definition:** En kontekstfri grammatik siges at være
 - **højre-regulær** hvis alle produktioner er på formen

$$A \rightarrow a \quad \text{eller} \quad A \rightarrow aB \quad \text{eller} \quad A \rightarrow \varepsilon$$

- **venstre-regulær** hvis alle produktioner er på formen

$$A \rightarrow a \quad \text{eller} \quad A \rightarrow Ba \quad \text{eller} \quad A \rightarrow \varepsilon$$

- **Sætning:** Et sprog er regulært
 - \Leftrightarrow det genereres af en højre-regulær grammatik
 - \Leftrightarrow det genereres af en venstre-regulær grammatik.
- Men højre og venstre må ikke blandes: Grammatikken

$$S \rightarrow aA \mid \varepsilon \quad A \rightarrow Sb$$

genererer $\{a^n b^n \mid n \in \mathbb{N}_0\}$!

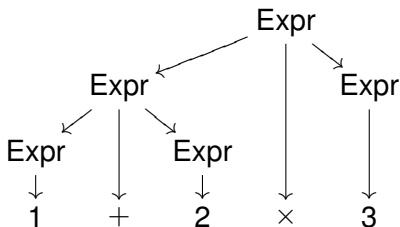
Kontekstfrie grammatikker og push-down-automater

- 4 Tvetydighed
- 5 Chomsky-normalformen
- 6 Push-down-automater
- 7 Ethvert kontekstfrit sprog genkendes af en PDA

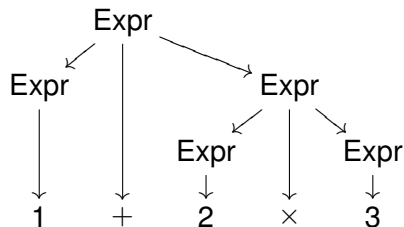
Eksempel: Grammatikken G_5 , ca.:

$\text{Expr} \rightarrow \text{Expr} + \text{Expr} \mid \text{Expr} \times \text{Expr} \mid (\text{Expr}) \mid \text{Heltal}$

To forskellige parsetræer for $1 + 2 \times 3$:



$= 9$



$= 7$

Definition: En derivation $S \Rightarrow w_1 \Rightarrow w_2 \Rightarrow \dots \Rightarrow w_k$ i en grammatik kaldes en **venstre-derivation** hvis det i ethvert skridt er den variable *længst til venstre* der erstattes.

Eksempel:

- $S \Rightarrow AB \Rightarrow aB \Rightarrow ab$ er en venstre-derivation,
- $S \Rightarrow AB \Rightarrow Ab \Rightarrow ab$ er ikke.

Bemærk: Til ethvert parsetræ svarer en entydig venstre-derivation.

Definition 2.7:

- Et ord siges at være genereret **tvetydigt** hvis det har to forskellige venstre-derivationer.
- En grammatik er **tvetydig** hvis den genererer et ord på en tvetydig måde.
- Et kontekstfrit sprog har en **iboende tvetydighed** hvis enhver CFG der genererer det er tvetydig.

Sætning: Der findes kontekstfrie sprog som har en iboende tvetydighed. (Opgave 2.29)

Sætning: Der findes ikke nogen algoritme som, givet en kontekstfri grammatik, kan afgøre om denne er tvetydig eller ej. (Opgave 5.21)

\Rightarrow i anvendelser: vigtigt at **designe** ikke-tvetydige CFGs

Mål: specielle former for kontekstfrie grammatikker som er nemme at håndtere

Definition 2.8: En CFG med startvariabel S er i **Chomsky-normalform** hvis hver produktion er af formen $A \rightarrow BC$ eller $A \rightarrow a$, hvor a er en terminal, A , B og C er variable og $B, C \neq S$. Desuden tillades produktionen $S \rightarrow \varepsilon$.

Sætning 2.9: Ethvert kontekstfrit sprog genereres af en CFG i Chomsky-normalform.

Bevis: Lad (V, Σ, R, S) være en CFG. Vi konverterer den til Chomsky-normalform:

❶ S må ikke forekomme på højresider.

Introducér en ny startvariabel S_0 og en produktion $S_0 \rightarrow S$.

Bevis: Lad (V, Σ, R, S) være en CFG. Vi konverterer den til Chomsky-normalform:

- ① S må ikke forekomme på højresider.
- ② Vi vil ikke have ε -produktioner $A \rightarrow \varepsilon$, medmindre $A = S$.
 - Tag en produktion $A \rightarrow \varepsilon$ og fjern den.
 - For alle produktioner $R \rightarrow uAv$: introducér en ny produktion $R \rightarrow uv$.
 - Men hvis der er en produktion $R \rightarrow A$, introduceres $R \rightarrow \varepsilon$ *kun hvis den ikke allerede før er blevet fjernet*.
 - Gentag indtil alle ε -produktioner er væk (undtaget måske $S \rightarrow \varepsilon$).

Bevis: Lad (V, Σ, R, S) være en CFG. Vi konverterer den til Chomsky-normalform:

- ❶ S må ikke forekomme på højresider.
- ❷ Vi vil ikke have ε -produktioner $A \rightarrow \varepsilon$, medmindre $A = S$.
- ❸ Vi vil ikke have *unit rules*: produktioner af formen $A \rightarrow B$.
 - Tag en produktion $A \rightarrow B$ og fjern den.
 - For alle produktioner $B \rightarrow u$: introducér en ny produktion $A \rightarrow u$.
 - Men hvis der er en produktion $B \rightarrow C$, introduceres $A \rightarrow C$ *kun hvis den ikke allerede før er blevet fjernet*.
 - Gentag indtil alle *unit rules* er væk.

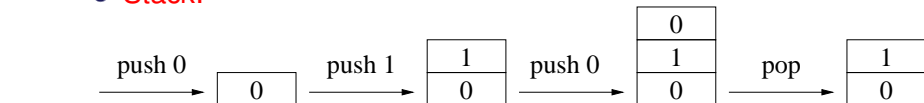
Bevis: Lad (V, Σ, R, S) være en CFG. Vi konverterer den til Chomsky-normalform:

- ① S må ikke forekomme på højresider.
- ② Vi vil ikke have ε -produktioner $A \rightarrow \varepsilon$, medmindre $A = S$.
- ③ Vi vil ikke have *unit rules*: produktioner af formen $A \rightarrow B$.
- ④ Vi vil ikke have produktioner af formen $A \rightarrow u_1 u_2 \dots u_k$ for $k \geq 3$.
 - Lad $A \rightarrow u_1 u_2 \dots u_k$ være en sådan produktion. (Her er u_i erne variable eller terminaler.)
 - Erstat den med produktioner $A \rightarrow u_1 A_1$, $A_1 \rightarrow u_2 A_2, \dots$, $A_{k-2} \rightarrow u_{k-1} u_k$, hvor A_i erne er nye variable.
 - Gentag.

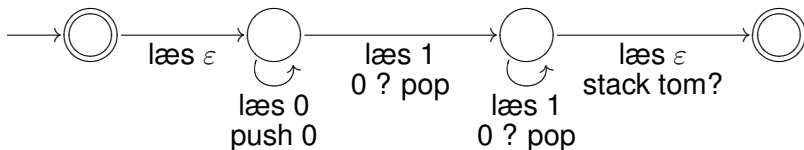
Bevis: Lad (V, Σ, R, S) være en CFG. Vi konverterer den til Chomsky-normalform:

- ① S må ikke forekomme på højresider.
- ② Vi vil ikke have ε -produktioner $A \rightarrow \varepsilon$, medmindre $A = S$.
- ③ Vi vil ikke have *unit rules*: produktioner af formen $A \rightarrow B$.
- ④ Vi vil ikke have produktioner af formen $A \rightarrow u_1 u_2 \dots u_k$ for $k \geq 3$.
- ⑤ Vi vil ikke have produktioner af formen $A \rightarrow bC$, $A \rightarrow Bc$ eller $A \rightarrow bc$.
 - Erstat $A \rightarrow bC$ med $A \rightarrow BC$ og $B \rightarrow b$, og gør lignende for de andre to. (Igen introduceres nye variable.)
- ⑥ Færdig!

- **Pushdown-automat:** endelig automat plus *stack*
- **Stack:**



- kan pushe symboler på stacken og læse og poppe det *øverste* stacksymbol
- Eksempel:



- genkender sproget $\{0^n 1^n \mid n \in \mathbb{N}_0\}$

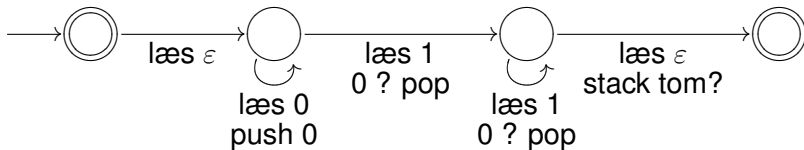
Definition 2.13: En **pushdown-automat (PDA)** er en 6-tupel $M = (Q, \Sigma, \Gamma, \delta, q_0, F)$, hvor delene er

- 1 Q : en endelig mængde af tilstande
- 2 Σ : input-alfabetet
- 3 Γ : stack-alfabetet
- 4 $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \rightarrow \mathcal{P}(Q \times \Gamma_\varepsilon)$: transitionsfunktionen
- 5 $q_0 \in Q$: starttilstanden
- 6 $F \subseteq Q$: mængden af accepttilstande

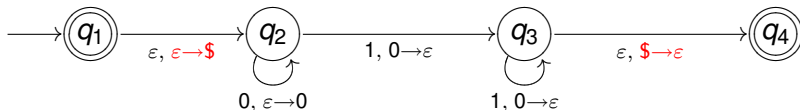
M siges at **acceptere** et ord $w \in \Sigma^*$ hvis der findes $m \in \mathbb{N}$ og $w_1, w_2, \dots, w_m \in \Sigma_\varepsilon$, $r_0, r_1, \dots, r_m \in Q$ og $s_0, s_1, \dots, s_m \in \Gamma^*$ således at $w = w_1 w_2 \dots w_m$ og

- 1 $r_0 = q_0$ og $s_0 = \varepsilon$,
- 2 for alle $i = 0, 1, \dots, m-1$ findes $a, b \in \Gamma_\varepsilon$ og $t \in \Gamma^*$ som opfylder $s_i = at$, $s_{i+1} = bt$ og $(r_{i+1}, b) \in \delta(r_i, w_{i+1}, a)$, og
- 3 $r_m \in F$.

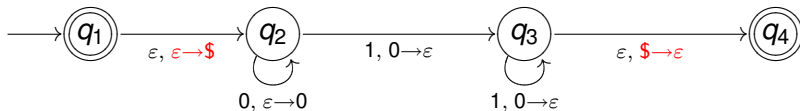
Eksempel 2.14:



At finde ud af om stacken er tom: Introducér et specielt
end-of-stack-symbol \$



Eksempel 2.14:



$$Q = \{q_1, q_2, q_3, q_4\}$$

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

$$\Gamma = \{0, \$\}$$

$$F = \{q_1, q_4\}$$

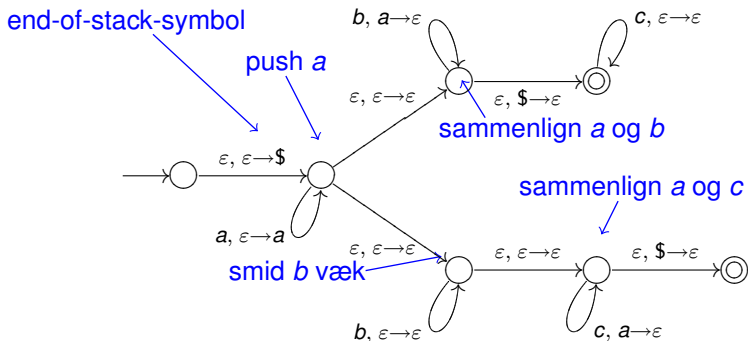
$\delta :$	Input:			0			1			ϵ		
	Stack:			0	\$	ϵ	0	\$	ϵ	0	\$	ϵ
q_1	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	$\{(q_2, \$)\}$
q_2	\emptyset	\emptyset	$\{(q_2, 0)\}$	$\{(q_3, \epsilon)\}$	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset
q_3	\emptyset	\emptyset	\emptyset	$\{(q_3, \epsilon)\}$	\emptyset	\emptyset	\emptyset	$\{(q_4, \epsilon)\}$	\emptyset	\emptyset	\emptyset	\emptyset
q_4	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

Opsummering: PDA:

- endelig automat med *stack*
- stacken kan gemme på *vilkårligt mange* symboler, men kun det *øverste* kan læses (og poppes)
- (*first-in, last-out*)
- *nondeterministiske*
- der findes deterministiske PDAs, ja. Men
 - vi skal ikke se på dem her, og
 - de genkender *færre* sprog end de nondeterministiske PDAs!

Eksempel 2.16: En PDA der genkender sproget

$$\{a^i b^j c^k \mid i, j, k \in \mathbb{N}_0 \text{ og } i = j \text{ eller } i = k\}$$



– det kan vises at man *skal* bruge en *nondeterministisk* PDA for at genkende det sprog

Lemma 2.21: Lad Σ være et alfabet og $A \subseteq \Sigma^*$ et kontekstfrit sprog. Da findes en PDA P med $\llbracket P \rrbracket = A$.

Bevis: Lad $G = (V, \Sigma, R, S)$ være en CFG med $\llbracket G \rrbracket = A$.

Idéen er at PDAen, givet en inputstreng s , nondeterministisk forsøger at finde en derivation for s i G :

- ➊ Push S på stacken
- ➋ Hvis topsymbolet på stacken er en variabel A : Pop A og push højresiden w af en produktion $A \rightarrow w$ i R . (Dø hvis der ikke er nogen produktion $A \rightarrow w$ i R .)
- ➌ Hvis topsymbolet på stacken er en terminal a : Sammenlign med næste inputsymbol. Hvis de er ens, pop a . Hvis de ikke er ens, dø.
- ➍ Gentag step 2 og 3 indtil stacken er tom.

Lemma 2.21: Lad Σ være et alfabet og $A \subseteq \Sigma^*$ et kontekstfrit sprog. Da findes en PDA P med $\llbracket P \rrbracket = A$.

Bevis: Lad $G = (V, \Sigma, R, S)$ være en CFG med $\llbracket G \rrbracket = A$.

Vi konstruerer først en “generaliseret PDA” $P = (Q, \Sigma, \Gamma, \delta, q_s, F)$, der kan pushe strenge i stedet for bare symboler. Lad

$Q = \{q_s, q_\ell, q_f\}$, $F = \{q_a\}$ og $\Gamma = V \cup \Sigma \cup \{\$ \}$. Lad

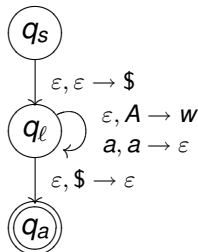
$$\delta(q_s, \varepsilon, \varepsilon) = \{(q_\ell, S\$)\}$$

$$\delta(q_\ell, \varepsilon, A) = \{(q_\ell, w) \mid w \in R(A)\} \quad \text{for alle } A \in V$$

$$\delta(q_\ell, a, a) = \{(q_\ell, \varepsilon)\} \quad \text{for alle } a \in \Sigma$$

$$\delta(q_\ell, \varepsilon, \$) = \{(q_a, \varepsilon)\}$$

$$\delta(q, a, b) = \emptyset \quad \text{for alle andre}$$



Lav til sidst P om til en “almindelig” PDA ved at erstatte enhver transition $q \xrightarrow{a, b \rightarrow s_1 s_2 \dots s_n} r$ med (nye tilstande og) en følge

$$q \xrightarrow{a, b \rightarrow s_n} q_1 \xrightarrow{\varepsilon, \varepsilon \rightarrow s_{n-1}} q_2 \longrightarrow \dots \longrightarrow q_{n-1} \xrightarrow{\varepsilon, \varepsilon \rightarrow s_1} r.$$