

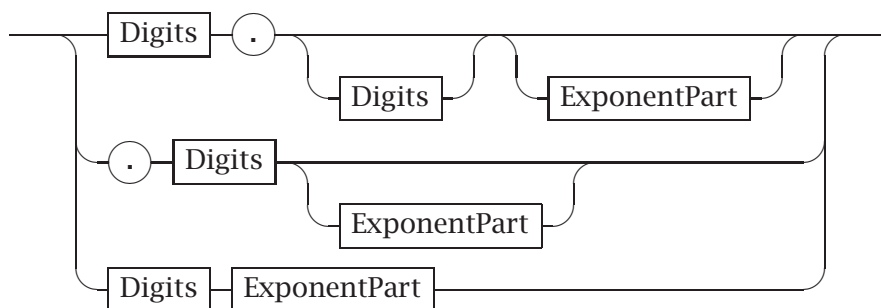
Løsningsforslag til ukeoppgaver i INF3110/4110

Uke 42 (15.-17.10.2003)

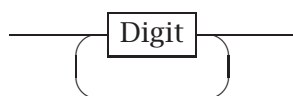
Oppgave 1

1. Jernbanediagram:

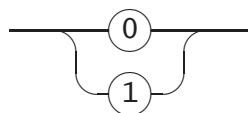
FloatingPointLiteral



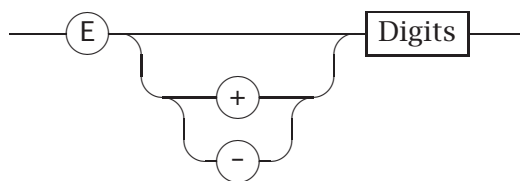
Digits



Digit



ExponentPart



2. Omskriving til regulær grammatikk

Én måte å omskrive definisjonen ser slik ut:

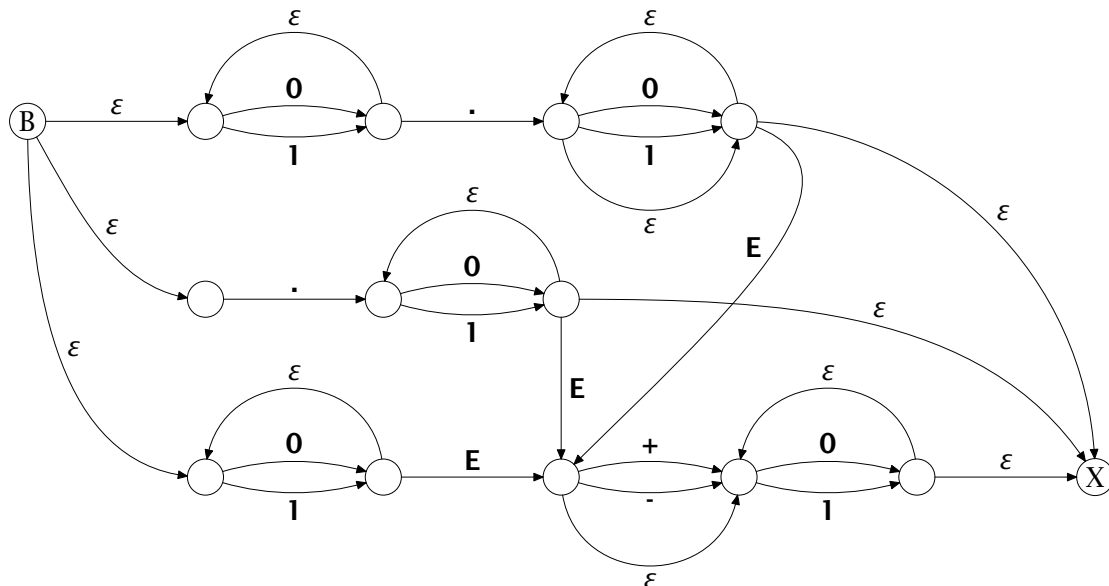
$$\begin{aligned}
 \langle fpl \rangle &\rightarrow \mathbf{0} \langle fpl1 \rangle \mid \mathbf{1} \langle fpl1 \rangle \mid \cdot \langle fpl2 \rangle \\
 \langle fpl1 \rangle &\rightarrow \mathbf{0} \langle fpl1 \rangle \mid \mathbf{1} \langle fpl1 \rangle \mid \cdot \langle fpl3 \rangle \mid \langle ep \rangle \\
 \langle fpl2 \rangle &\rightarrow \mathbf{0} \langle fpl3 \rangle \mid \mathbf{1} \langle fpl3 \rangle \\
 \langle fpl3 \rangle &\rightarrow \mathbf{0} \langle fpl3 \rangle \mid \mathbf{1} \langle fpl3 \rangle \mid \langle ep \rangle \mid \varepsilon \\
 \\
 \langle ep \rangle &\rightarrow \mathbf{E} \langle ep1 \rangle \\
 \langle ep1 \rangle &\rightarrow + \langle ep2 \rangle \mid - \langle ep2 \rangle \mid \langle ep2 \rangle \\
 \langle ep2 \rangle &\rightarrow \langle digits \rangle \\
 \\
 \langle digits \rangle &\rightarrow \mathbf{0} \mid \mathbf{1} \mid \mathbf{0} \langle digits \rangle \mid \mathbf{1} \langle digits \rangle
 \end{aligned}$$

Tanken bak metasymbolene er:

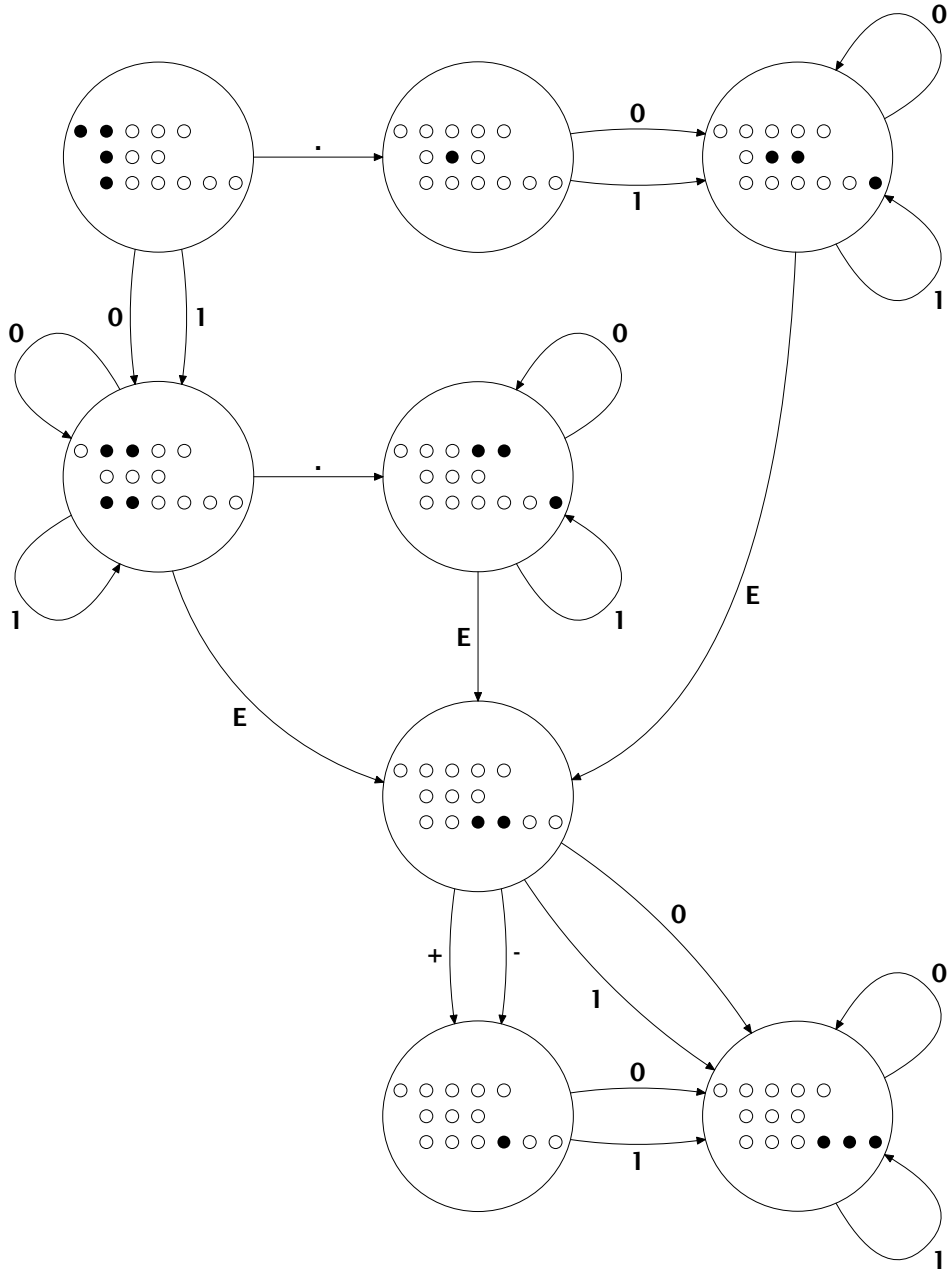
- $\langle fpl \rangle$ er $\langle FloatingPointLiteral \rangle$.
- $\langle fpl1 \rangle$ er 1. og 3. alternativ av $\langle fpl \rangle$ etter at vi har lest minst ett siffer.
- $\langle fpl2 \rangle$ er 2. alternativ av $\langle fpl \rangle$ etter at vi har lest punktumet.
- $\langle fpl3 \rangle$ er situasjonen for alle alternativene av $\langle fpl \rangle$ etter at punktum er lest.
- $\langle ep \rangle$ er $\langle ExponentPart \rangle$.
- $\langle ep1 \rangle$ er situasjonen etter at **E** i $\langle ep \rangle$ er lest.
- $\langle ep2 \rangle$ er etter at **E** og eventuelt fortegn i $\langle ep \rangle$ er lest.

3. Fra jernbanediagram til ikke-deterministisk automat

Én mulig ID-automat er vist Figur 1. Den er laget nesten «rett frem», men for at ikke automaten skulle bli for stor, er det ikke laget flere kopier av $\langle ExponentPart \rangle$. (Starttilstanden er her merket "B").



Figur 1: En ID-automat for $\langle FloatingPointLiteral \rangle$



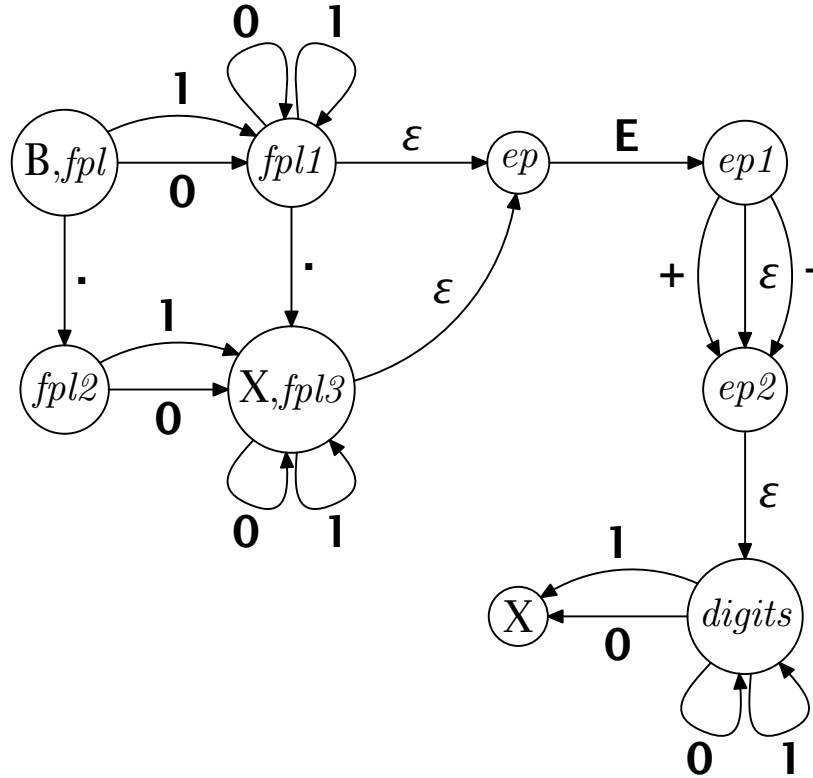
Figur 2: En D-automat for $\langle \text{FloatingPointLiteral} \rangle$

4. Tilsvarende deterministisk automat

Den deterministiske utgaven av ID-automaten i Figur 1 er vist i Figur 2. Denne figuren viser automaten tegnet slik det er gjort i kompendiet. Etter «ren-tegning» ser den ut som vist i Figur 3.

5. Fra BNF-grammatikk til ikke-deterministisk automat

Den «naturlige» automaten ut fra grammatikken fra punkt 2 er vist i Figur 4.



Figur 4: En ID-automat for $\langle fpl \rangle$

6. Tilsvarende deterministisk automat

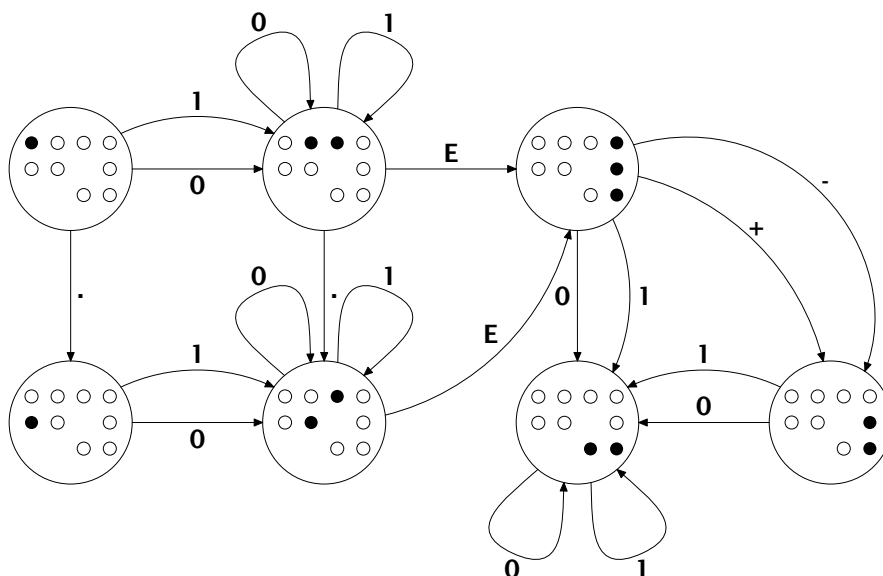
Den deterministiske utgaven av ID-automaten i Figur 4 er vist i Figur 5.

Tilstandstabell

Tabellen for automaten i Figur 5 er vist i Tabell 1.

Tilstand	0	1	.	+	-	E	
1 = FPL	3	3	2	F	F	F	
2 = FPL2	4	4	F	F	F	F	
3 = FPL1, EP	3	3	4	F	F	5	
4 = FPL3, EP	4	4	F	F	F	5	JA
5 = EP1, DIGITS, EP2	6	6	F	7	7	F	
6 = DIGITS	6	6	F	F	F	F	JA
7 = EP2, DIGITS	6	6	F	F	F	F	
F (= "FEIL")	F	F	F	F	F	F	

Tabell 1: Tilstandstabell for $\langle fpl \rangle$



Figur 5: En D-automat for $\langle fpl \rangle$

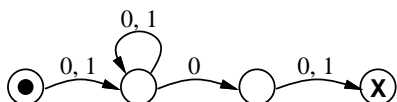
7. Sammenligning

Automaten i Figur 3 inneholder en tilstand mer enn automaten i Figur 5. Vi ser imidlertid at tilstanden øverst til høyre i Figur 3 kan slå sammen med tilstanden under, da begge har helt like utkanter (samme symbol til samme node) og begge er slutttilstander. Etter å ha gjort en slik sammenslåing vil vi da ha samme automat som i Figur 3. En deterministisk automat hvor alle mulige sammenslåinger er gjort, kalles *minimal*. Ethvert regulært språk har en entydig minimal D-automat.

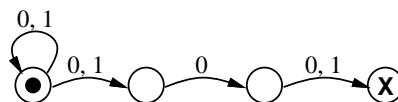
Oppgave 2

1. Setningene er vilkårlige sekvenser av 1'ere og 0'ere, men slik at sekvensen minst er tre tegn lang og slik at det nest siste tegnet i sekvensen er 0.
2. To mulige ID-automater er:

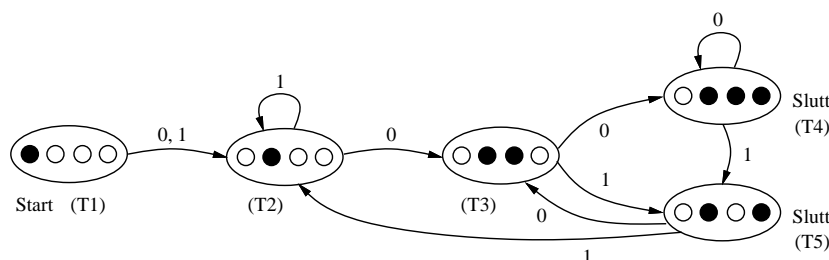
ID-automat nr. 1:



ID-automat nr. 2:



3. Med utgangspunkt i ID-automat nr. 1 får vi følgende D-automat:



4. Her er en regulær grammatikk som beskriver språket:

$$\begin{aligned}\langle A \rangle &\rightarrow \mathbf{0} \langle B \rangle \mid \mathbf{1} \langle B \rangle \\ \langle B \rangle &\rightarrow \mathbf{0} \langle B \rangle \mid \mathbf{1} \langle B \rangle \mid \mathbf{0} \langle C \rangle \\ \langle C \rangle &\rightarrow \mathbf{0} \mid \mathbf{1}\end{aligned}$$

Oppgave 3

1. En avledning av setningen **3333555555**:

$$\begin{aligned}\langle S \rangle &\Rightarrow \langle A \rangle \langle B \rangle \\ &\Rightarrow \mathbf{3} \langle A \rangle \mathbf{3} \langle B \rangle \\ &\Rightarrow \mathbf{3} \mathbf{3} \langle A \rangle \mathbf{3} \mathbf{3} \langle B \rangle \\ &\Rightarrow \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{3} \langle B \rangle \\ &\Rightarrow \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{5} \langle B \rangle \mathbf{5} \\ &\Rightarrow \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{5} \mathbf{5} \langle B \rangle \mathbf{5} \mathbf{5} \\ &\Rightarrow \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{5} \mathbf{5} \mathbf{5} \langle B \rangle \mathbf{5} \mathbf{5} \mathbf{5} \\ &\Rightarrow \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{3} \mathbf{5} \mathbf{5} \mathbf{5} \mathbf{5} \mathbf{5}\end{aligned}$$

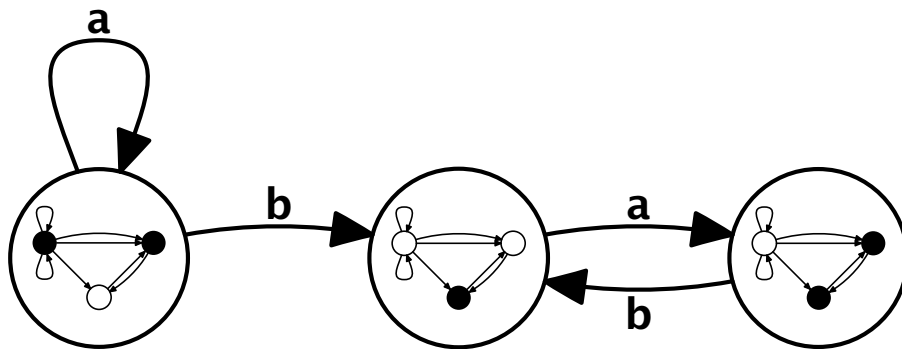
2. Setningene i dette språket er strenger som begynner et *like* antall **3**'ere og som slutter med et *like* antall **5**'ere.

Oppgave 4

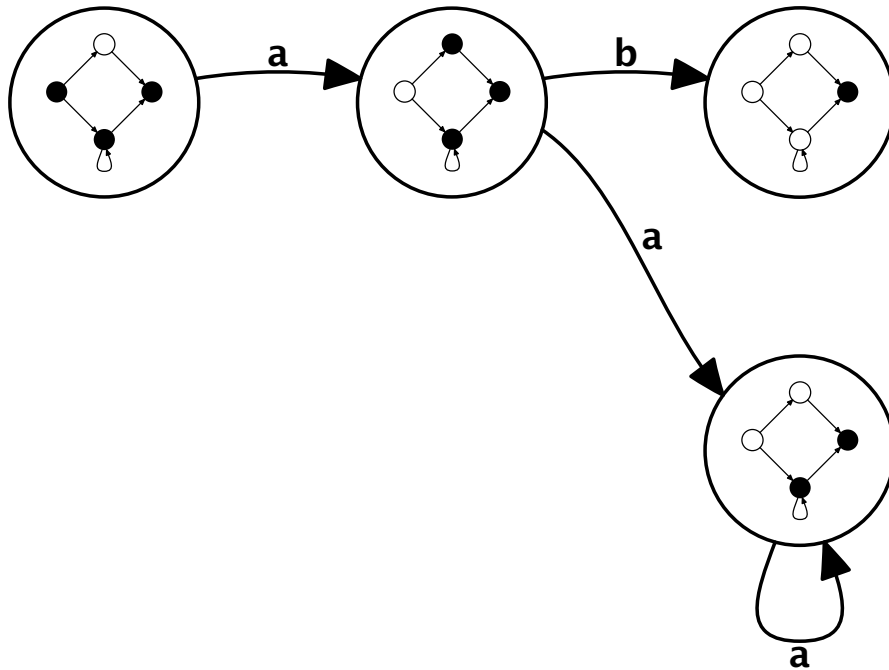
1. **abb**: JA (B, 1, 2, 3, X)
2. **abbbb**: NEI
3. **aabb**: JA (B, 4, 5, 5, 5, X)
4. **abab**: JA (B, 1, 2, 3, 3, X)
5. **ababb**: NEI
6. **ababab**: JA: (B, 1, 2, 1, 2, 3, 3, X)

Oppgave 5

I denne automaten er de to tilstandene lengst til høyre slutttilstander; tilstanden lengst til venstre er starttilstanden.



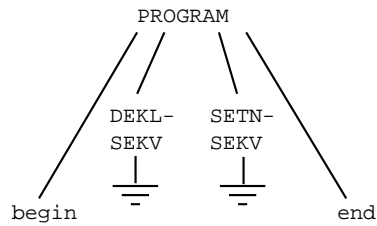
I denne automaten er alle tilstandene slutttilstander; tilstanden lengst til venstre er starttilstanden.



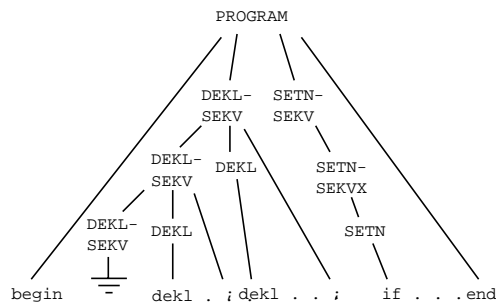
Oppgave 6 (kan puffes til neste uke)

1. To eksempler:

Setningen `begin end` har syntakstree:



Setningen `begin dekl ... ; dekl ... ; if ... end` har syntakstree:



2. $\langle Deklsekv \rangle$ og $\langle Setnsekv \rangle$ er de eneste som kan produsere den tomme setning.

3. Startmengder:

Metasymbol	Startmengde	
$\langle Program \rangle$	begin	
$\langle Deklsekv \rangle$		dekl
$\langle Setnsekv \rangle$		put if while
$\langle SetnsekvX \rangle$		put if while
$\langle Dekl \rangle$	dekl	
$\langle Setn \rangle$	put if while	