Løsningsforslag til hjemmeeksamen 1 i INF3110/4110

Roger Antonsen

Oktober 2003

Del 1 - Programmering i ML

Det er *mange* gode måter å løse denne oppgaven på. Her er et minimalistisk forslag med en del bruk av høyere-ordens funksjoner.

```
datatype 'a tre = Node of 'a | Tre of 'a * ('a tre list);

(* Noen hjelpefunksjoner *)
fun max2 (x, y) = if x > y then x else y;
fun max xs = foldr max2 0 xs;
fun sum xs = foldr op+ 0 xs;
```

(1 poeng) Oppgave 1 `dybde : 'a tre -> int`

```
fun dybde (Node(_)) = 0
  | dybde (Tre(_, [])) = 0
  | dybde (Tre(_, ts)) = 1 + max (map dybde ts);
```

(1 poeng) Oppgave 2 `forgrening : 'a tre -> int`

```
fun forgrening (Node(_)) = 0
  | forgrening (Tre(_, ts)) = max2 ((length ts), max (map forgrening ts));
```

(1 poeng) Oppgave 3 `antall : 'a tre -> int`

```
fun antall (Node(_)) = 1
  | antall (Tre(_, ts)) = 1 + sum (map antall ts);
```

(1 poeng) **Oppgave 4** `flatD : 'a tre -> 'a list`

```
fun flatD (Node(x)) = [x]
  | flatD (Tre(x, ts)) = x :: (foldr op@ [] (map flatD ts));
```

(2 poeng) **Oppgave 5** `flatB : 'a tre -> 'a list`

```
fun flatBlist [] = []
  | flatBlist (Node(x) :: rest) = x :: flatBlist rest
  | flatBlist (Tre(x, ts) :: rest) = x :: flatBlist(rest @ ts);

fun flatB x = flatBlist [x];
```

(2 poeng) **Oppgave 6** `trelik : ”a tre -> ”a tre -> bool`

```
fun alletrelik [] [] = true
  | alletrelik (x::xs) (y::ys) = trelik x y andalso alletrelik xs ys
  | alletrelik _ _ = false
and trelik (Node(x)) (Node(y)) = x = y
  | trelik (Tre(x, xs)) (Tre(y, ys)) = (x = y) andalso alletrelik xs ys
  | trelik _ _ = false;
```

(2 poeng) **Oppgave 7** `speil : 'a tre -> 'a tre`

```
fun speil (Node(x)) = Node(x)
  | speil (Tre(x, xs)) = Tre(x, map speil (rev xs));
```

Del 2 - Syntaks

(3 poeng) **Oppgave 8**

Kommentarer

- Ca. 0,5 poeng for hver deloppgave.
- Grammatikkene må generere *alle* og *bare* gyldige strenger.
- Den må ha et øverste metasymbol.
- Den kan ikke tillate tomme strenger.
- Produksjoner av typen $\langle E \rangle \to \langle E \rangle$ er unødvendige.
- Følgende strenger *må* godtas: **3**, **321**, **32**, **31**.

a)

Løsningsforslag

Det er mange gode løsninger. Her er to forslag:

$\langle T\rangle \rightarrow \mathbf{3}\ \langle TRE\rangle \mid \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle$
$\langle TRE\rangle \rightarrow \mathbf{3}\ \langle TRE\rangle \mid \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle \mid \varepsilon$
$\langle TO\rangle \rightarrow \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle \mid \varepsilon$
$\langle EN\rangle \rightarrow \mathbf{1}\ \langle EN\rangle \mid \varepsilon$

$\langle T\rangle \rightarrow \mathbf{3} \mid \mathbf{3}\ \langle T\rangle \mid \langle TO\rangle$
$\langle TO\rangle \rightarrow \mathbf{2} \mid \mathbf{2}\ \langle TO\rangle \mid \langle EN\rangle$
$\langle EN\rangle \rightarrow \mathbf{1} \mid \mathbf{1}\ \langle EN\rangle$

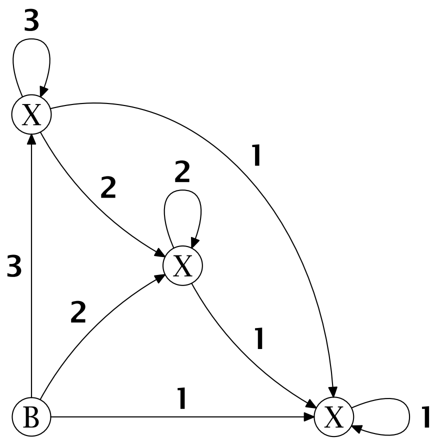
b)

Løsningsforslag

$\langle T\rangle \rightarrow \mathbf{3}^{+}\ \mathbf{2}^{*}\ \mathbf{1}^{*} \mid \mathbf{2}^{+}\ \mathbf{1}^{*} \mid \mathbf{1}^{+}$

c)

Løsningsforslag



Kommentarer

- Den må godta **31**.
- Den må være deterministisk.

d)

Løsningsforslag

Her det også mange gode løsninger. Her er to forslag:

$\langle T\rangle \rightarrow \mathbf{3}\ \langle TRE\rangle \mid \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle$
$\langle TRE\rangle \rightarrow \mathbf{3}\ \langle TRE\rangle \mid \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle \mid \mathbf{+}\ \langle T\rangle \mid \varepsilon$
$\langle TO\rangle \rightarrow \mathbf{2}\ \langle TO\rangle \mid \mathbf{1}\ \langle EN\rangle \mid \mathbf{+}\ \langle T\rangle \mid \varepsilon$
$\langle EN\rangle \rightarrow \mathbf{1}\ \langle EN\rangle \mid \mathbf{+}\ \langle T\rangle \mid \varepsilon$

$\langle T\rangle \rightarrow \mathbf{3} \mid \mathbf{3}\ \langle T\rangle \mid \langle TO\rangle \mid \mathbf{3}\ \mathbf{+}\ \langle T\rangle$
$\langle TO\rangle \rightarrow \mathbf{2} \mid \mathbf{2}\ \langle TO\rangle \mid \langle EN\rangle \mid \mathbf{2}\ \mathbf{+}\ \langle T\rangle$
$\langle EN\rangle \rightarrow \mathbf{1} \mid \mathbf{1}\ \langle EN\rangle \mid \mathbf{1}\ \mathbf{+}\ \langle T\rangle$

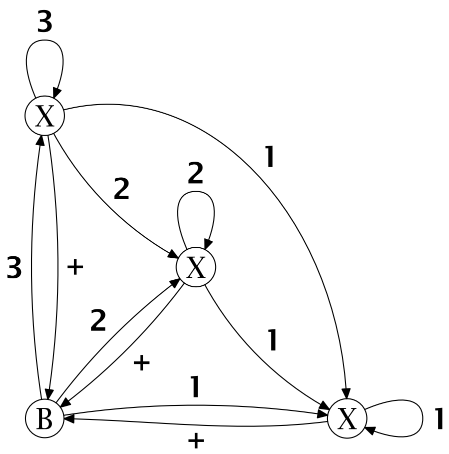
Kommentarer

- Den kan ikke godta flere **+**'er etter hverandre.
- Strenger hvor **+** er siste symbol er ikke tillatt.

e) Løsningsforslag

$\langle T\rangle \rightarrow [\![\mathbf{3}^+\ \mathbf{2}^*\ \mathbf{1}^* \mid \mathbf{2}^+\ \mathbf{1}^* \mid \mathbf{1}^+]\!]\ [\![\mathbf{+}\ [\![\mathbf{3}^+\ \mathbf{2}^*\ \mathbf{1}^* \mid \mathbf{2}^+\ \mathbf{1}^* \mid \mathbf{1}^+]\!]\ ]\!]^*$

f) Løsningsforslag



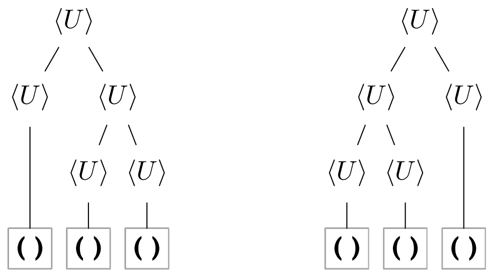
(2 poeng) **Oppgave 9**

Kommentarer

- Ca. 1 poeng for (a) og (b) tilsammen, ca. 1 poeng for (c).

a) Løsningsforslag

Strengen **()()()** har to syntakstrær; derfor er grammatikken flertydig.



b)

Løsningsforslag

Den er *ikke* regulær, siden det er en produksjon

$\langle U \rangle \rightarrow \mathbf{(}\langle U \rangle\mathbf{)}$

som har tegnet **)** til sist. For at grammatikken skal være regulær, så kan det være maksimalt ett metasymbol på høyresiden i en gitt produksjon og det må forekomme sist.

Kommentarer

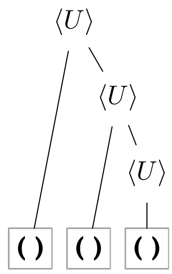
- "Maks ett metasymbol på høyresiden" gjelder *for hver produksjon*.

c)

Løsningsforslag

$\langle U \rangle \rightarrow \mathbf{()} \mid \mathbf{()}\ \langle U \rangle \mid \mathbf{(}\langle U \rangle\mathbf{)} \mid \mathbf{(}\langle U \rangle\mathbf{)}\ \langle U \rangle$

Som et eksempel vil syntakstreet for **()()()** nå være entydig bestemt:



Kommentarer

- Den skal være entydig. Sjekk f.eks. for **()()()** og **()(())**. Følgende forslag er f.eks. ikke tilstrekkelig for å garantere entydighet (Man får fortsatt to syntakstrær for **()()()**.)

  $\langle U \rangle \rightarrow \mathbf{()} \mid \mathbf{()}\ \langle U \rangle \mid \langle U \rangle\langle V \rangle$
  $\langle V \rangle \rightarrow \langle U \rangle$

- Den må fortsatt generere *alle* setningene fra språket. (Det er ikke nok at den er entydig.)
- Eksempelvis må følgende være gyldige strenger: **(())**, **(()())**, **()(())**, **()()()**, **(())()**.

(3 poeng) **Oppgave 10**

Kommentarer

- Ca. 1 poeng for (a) og riktig svar for (b). God begrunnelse for (b) gir 1 poeng til.

a)

Løsningsforslag

Det er også mange måter å løse denne på. Her er to:

$\langle S \rangle \rightarrow \mathbf{a}\ \langle b\text{-}løfte \rangle \mid \mathbf{b}\ \langle a\text{-}løfte \rangle \mid \varepsilon$
$\langle b\text{-}løfte \rangle \rightarrow \mathbf{b}\ \langle S \rangle \mid \mathbf{a}\ \langle b\text{-}løfte \rangle\ \langle b\text{-}løfte \rangle$
$\langle a\text{-}løfte \rangle \rightarrow \mathbf{a}\ \langle S \rangle \mid \mathbf{b}\ \langle a\text{-}løfte \rangle\ \langle a\text{-}løfte \rangle$

Idéen her er følgende: hvis en streng begynner med en **a**, så må vi sørge for at det kommer en **b** utover i strengen; vi bruker ⟨*b-løfte*⟩ til å representere et slikt "løfte". En ⟨*b-løfte*⟩ kan enten innfris ved at det kommer en **b**, hvorpå vi kan legge til enda en ⟨*S*⟩, som har like mange **a**'er som **b**'er, eller, hvis man leser ytterligere en **a**, så må man sørge for å få med *et ekstra* ⟨*b-løfte*⟩. TIlsvarende for **b**.

$\langle S \rangle \rightarrow \mathbf{a}\ \langle S \rangle\ \mathbf{b}\ \langle S \rangle \mid \mathbf{b}\ \langle S \rangle\ \mathbf{a}\ \langle S \rangle \mid \varepsilon$

Idéen med denne er tilsvarende.

Kommentarer

- Strenger må kunne både begynne og slutte med **a**'er. Her er det lett å tenke riktig, men ikke grundig nok. F.eks. må **aabbba** og **aabbbbaa** være tillatt.
- Grammatikken må ikke tillate for mye.
- Grammatikken må ikke tillate for lite, f.eks. bare $\mathbf{a}^n\mathbf{b}^n$.

b) Løsningsforslag

Det er ikke mulig å gi en regulær grammatikk for dette språket. Vi har ikke gjennomgått nok i dette kurset for kunne gi et matematisk *presist* argument for dette, men en skisse til bevis/forklaring er følgende. Hvis vi hadde kunnet lage en regulær grammatikk for dette språket, så hadde det eksistert en *endelig* automat som gjenkjente nøyaktig de gyldige strengene. Denne automaten måtte ha vært i stand til å *telle* antall **a**'er for å være i stand til å generere like mange **b**'er. Siden det ikke er gitt noen øvre grense for hvor mange **a**'er som kan komme før tilsvarende **b**'er, så kan det ikke eksistere noen *endelig* automat som gjør dette. (Anta at automaten har n tilstander. Merk hver tilstand med det antallet **a**'er som kan være lest til nå. Det må da finnes en tilstand som er er merket *forskjellig* antall **a**'er. Hvordan kan automaten da garantere at det kommer et korrekt antall **b**'er derfra?) Derfor kan språket heller ikke utrykket som en regulær grammatikk.

Kommentarer

- Det bør være med noe om endelig/uendelig/lagerplass/antall symboler, etc.
- Det er ikke tilstrekkelig å si at en regulær grammatikk ikke kan ha to metasymboler på høyresiden.

(2 poeng) Oppgave 11

a) Løsningsforslag

Denne automaten gjenkjenner alle endelig sekvenser av 0 og 1 som ender med to 0'er etter hverandre.

b) Løsningsforslag

$\langle B \rangle \rightarrow [\![\mathbf{0} | \mathbf{1}]\!]^{*}\mathbf{00}$