

INF5110

Oppgaver som gjennomgås

7. februar 2014

Spørsmålene på de tre neste foilene, samt:

- Finn *First*- og *Follow*-mengdene til grammatikken i eksempelet på side 160 *etter at* venstre-rekursjon er fjernet.
- Beskriv en algoritme som bare finner de utnullbare ikke-terminaler, uten å beregne First og Follow.



Oppgave: Entydighet uten presedens-regler etc.

Om vi har operasjoner $+$ $-$ $*$ $/$ og \uparrow (opphøying) så kan vi få entydighet med følgende regler:

- $+$, $-$ lav, venstre-ass.
- $*$, $/$ høyere, venstre-ass.
- \uparrow , høyest, høyre-ass.

Om vi skriver:

$$3 + 5 / 3 * 2 + 4 \uparrow 2 \uparrow 3$$

så vil dette da bli tolket som:

$$(3 + ((5 / 3) * 2)) + (4 \uparrow (2 \uparrow 3))$$

Oppgave: Skriv en entydeig grammatikk som gir tolkningen over uten å måtte bruke ekstra regler.

Noen oppgaver som gjennomgås 5. februar

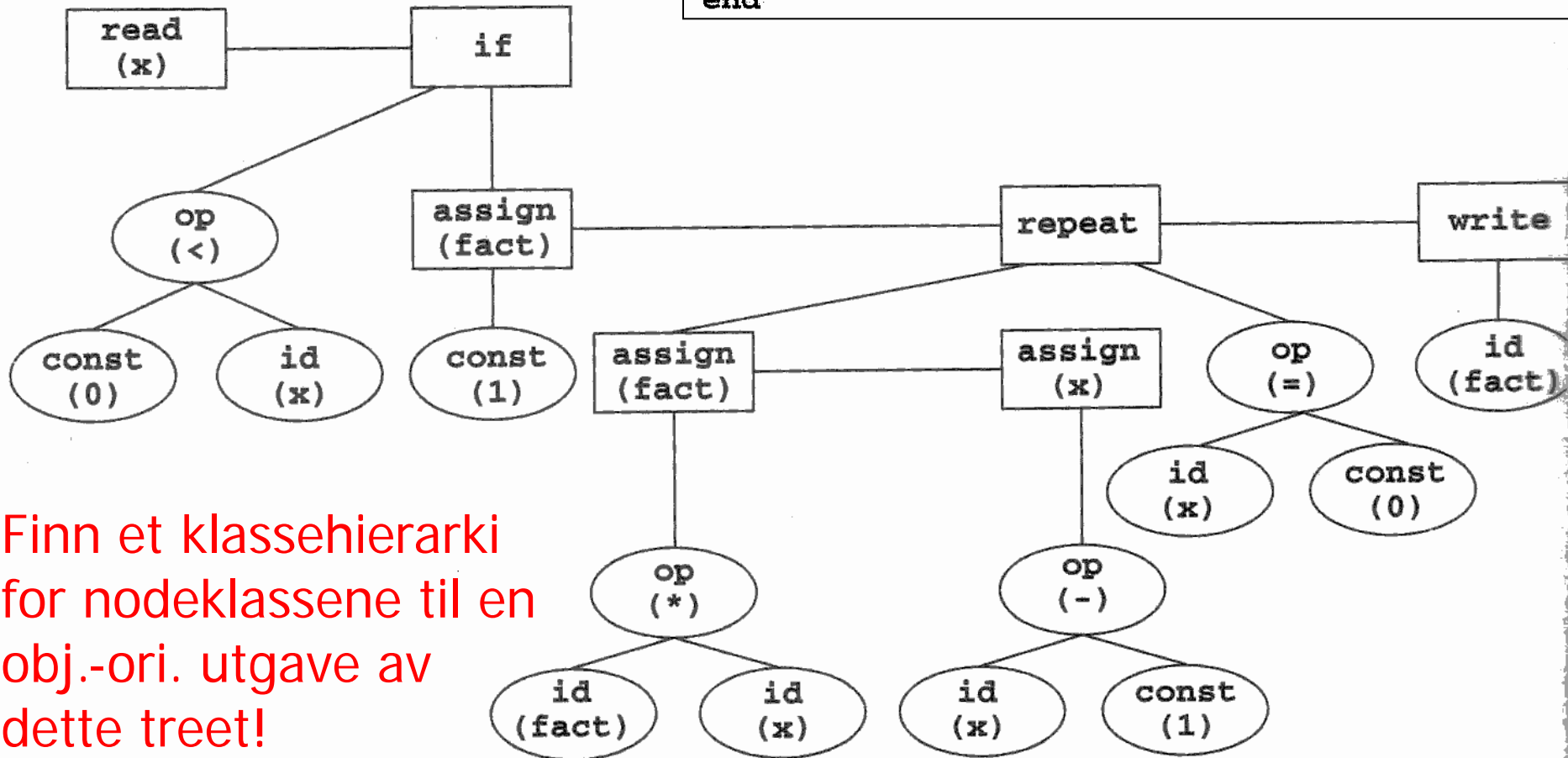
program → *stmt-sequence*
stmt-sequence → *stmt-sequence* ; *statement* | *statement*
statement → *if-stmt* | *repeat-stmt* | *assign-stmt* | *read-stmt* | *write-stmt*
if-stmt → **if** *exp* **then** *stmt-sequence* **end**
 | **if** *exp* **then** *stmt-sequence* **else** *stmt-sequence* **end**
repeat-stmt → **repeat** *stmt-sequence* **until** *exp*
assign-stmt → **identifier** := *exp*
read-stmt → **read** *identifier*
write-stmt → **write** *exp*
exp → *simple-exp* *comparison-op* *simple-exp* | *simple-exp*
comparison-op → < | =
simple-exp → *simple-exp* *addop* *term* | *term*
addop → + | -
term → *term* *mulop* *factor* | *factor*
mulop → * | /
factor → (*exp*) | **number** | **identifier**

- Er grammatikken entydig?
- Hva om vi vil tillate tomme setninger
- Hva om vi vill ha semikolon etter og ikke mellom setningene?
- Hva slags assosiativitet og presedens er det for operatorene?

Oppgave til 5. februar

Abstrakt syntakstre for
Tiny-programmet:

```
read x; { input an integer }
if 0 < x then { don't compute if x <= 0 }
  fact := 1;
  repeat
    fact := fact * x;
    x := x - 1
  until x = 0;
write fact { output factorial of x }
end
```



Finn et klassehierarki
for nodeklassene til en
obj.-ori. utgave av
dette treet!