

# Oppgaver INF 2820 V2011

## Gruppe 17.2

Det er to typer oppgaver. Det er en praktisk oppgave som inngår i innleveringen oblig 1b. Denne kan dere arbeide med på gruppa og få råd og vink.

I tillegg er det teorioppgaver. Disse gir trening i begreper fra forelesningene og vil være eksamensforberedende. Her kan det være klokt å prøve å arbeide med oppgavene også på egen hånd før gruppa.

## Teorioppgaver

### Oppgave 1:

På forelesningen skal vi se at språket der  $A = \{a, b, c\}$ ,  $L = \{a^n b^n c^n \mid n \geq 0\}$  ikke er kontekstfritt.

La  $L_2$  være komplementet til  $L$ , mao  $L_2 = \{w \in A^* \mid w \neq a^n b^n c^n \text{ for alle } n \geq 0\}$ . Vis at  $L_2$  er kontekstfritt. (Hint.  $L_2$  er unionen av flere kontekstfrie språk: Alle uttrykk som ikke er på formen  $a^* b^* c^*$ , de som er på formen  $a^n b^k c^m$  der  $m \neq n$ , osv.)

Det følger at komplementet til et kontekstfritt språk ikke behøver å være det. Sammenlikn dette med regulære språk.

### Oppgave 2:

Vis at språket der  $A = \{a, b, c\}$ ,  $L = \{a^n b^n c^n \mid n \geq 0\}$  kan skrives som snittet av to kontekstfrie språk. Det følger at kontekstfrie språk ikke er lukket under snitt.

### Oppgave 3:

På forelesningen sa vi at språket beskrevet av en venstrelinjær grammatikk er regulært, og det samme er tilfelle for en høyrelinjær grammatikk. Vis at en grammatikk hvor hver enkelt regel er høyrelinjær eller venstrelinjær (eller med andre ord en grammatikk hvor hver regel inneholder toppen en ikke-terminal på høyresiden og denne ikke-terminalen står enten lengst til høyre eller lengst til venstre i høyresiden) ikke behøver å være regulær. (Hint ta en CFG for et språk du vet er kontekstfritt, men ikke regulært, og prøv å omform grammatikken til denne formen.)

## Praktisk oppgave, del av oblig 1b

### Oppgave 4:

Vi skal arbeide med kontekstfrie grammatikker (CFGs). Vi skal bruke et hjelpemiddel fra NLTK, the Natural Language Toolkit. Dette er en samling av programmer for å lære datalingvistikk. Programmene er skrevet i Python. Det behøver ikke vi ta hensyn til.

På en av UiOs Linux-terminaler gi kommandoen: idle

Du kommer nå inn i en pythonsesjon med prompt >>>

Skriv

```
>>> import nltk
```

```
>>> nltk.app.rdparser()
```

Du skal nå få opp et vindu med en grammatikk på den ene siden og plass for trær på den andre siden.

- a) Finn ut hvilke trær grammatikken tilordner til eksempelsetningen. Lagr trærne, enten ved å bruke applikasjonens Print to postscript, eller et annet hjelpemiddel. Trærne skal leveres inn.
- b) Vi skal gjøre et par endringer til grammatikken. Fjern reglene:
  - a. NP → Det N PP
  - b. VP → V NP PP

Legg til reglene

- c. NP → NP PP
- d. VP → VP PP

Finn trærne som denne grammatikken tilordner til eksempelsetningen. Trærne skal leveres inn.

- c) Ta nå eksempelsetningen
  - a. The dog saw a man in the park with a telescope.Tegn opp alle trær denne får ut i fra grammatikken. Ta vare på dem som over og lever inn.
- d) Hvorfor tror du den opprinnelige eksempelgrammatikken brukte andre regler, regler som er dårligere lingvistisk motivert? (Hint: Bruk Autostep med de to grammatikkene. Sitt klar med ctrl-c.) Bruk noen setninger på å svare på spørsmålet.