

# Grundläggande logik och modellteori

## Höstterminen 2014

### Laboration 1: Prolog-uppvärmning och satslogik

**Deadline:** 2014-09-12, 12:00

**Inlämning:** Skicka din lösning till Niklas Zechner (zechner@cs.umu.se). Ärendemeningen ska vara

[5dv102] namn-1

där 'namn' är ditt cs-användarnamn. Din *väl kommenterade* lösning ska finnas i en bifogad fil som heter

namn-1.pl

där 'namn' är ditt användarnamn. Glöm inte att klistra in dina quiz-svar i en kommentar.

## Att börja med Prolog

### Tekniska detaljer

För kursen använder vi SWI-Prolog, som finns tillgänglig på <http://www.swi-prolog.org/>. Vi kommer att testa era lösningar på Datavetenskaps Linux-maskiner, med den version av SWI-Prolog som finns installerad där. För att testköra era lösningar på Datavetenskap:

1. Logga in på en av Linux-maskinerna (använd `ssh` eller logga in direkt).
2. Starta ett terminalfönster och se till att er lösningsfil finns i nuvarande katalog.
3. Starta SWI-Prolog med kommandot `swipl`.
4. Ladda er fil genom att skriva `consult (filnamn) .`
5. Testa att köra de queries som uppgiften kräver.
6. Kom ihåg att svarssträngen från frågorna som skall besvaras också skall placeras som en kommentar högst upp i Prolog-filen som del av inlämningen!

### Att lära sig Prolog

Gå igenom den Prolog-tutorial som finns på [http://www.doc.gold.ac.uk/~mas02gw/prolog\\_tutorial/prologpages/](http://www.doc.gold.ac.uk/~mas02gw/prolog_tutorial/prologpages/), den innehåller allt ni behöver veta för den här uppgiften.

### Att interagera med Prolog

På Prolog-prompten använder ni `consult/1` för att ladda filen med era predikat. Ni skriver sedan en "query", en fråga till Prolog. Testa t.ex. frågan `member(X, [1, 2, 3])`. Prolog kommer att svara `X = 1 ?`, eftersom X är en medlem i listan `[1, 2, 3]` om `X=1`. Trycker ni på ";" här kommer Prolog att ge ett förslag till på lösning, `X = 2 ?`, detta kan upprepas tills Prolog givit alla möjliga svar. Om man är nöjd med svaret trycker man bara på retur för att återgå till prompten.

Testa också frågan  $\text{member}(X, [1, 2, 3]), \text{member}(Y, [1, 2, 3]), X \neq Y$ . Komman i Prolog-frågor betyder “och”, så detta frågar efter ett X och ett Y som båda är medlemmar av listan  $[1, 2, 3]$  men som är olika ( $\neq$  är Prologs inte-lika-med-operator). Prolog ger total 6 olika svar på den frågan.

## Uppgiften

Denna uppgift är om symptom och sjukdomar. Vi vet följande:

- Peter har influensa.
- Peter har gulsot.
- John har gulsot.
- Charles har matförgiftning.
- Mary har influensa.
- Trötthet är ett symptom på gulsot.
- Diarré är ett symptom på matförgiftning.
- Feber är ett symptom på influensa.
- Trötthet är ett symptom på influensa.
- Aspirin lindrar feber.
- Lomofil hjälper mot diarré.

Vi har också följande kunskap:

- En medicin lindrar en sjukdom om sjukdomen har symptom som elimineras av medicinen.
- En person ska ta medicin om personen har en sjukdom som lindras av medicinen.

Skriv ett Prolog-program som tar hänsyn till de direkta fakta och de indirekta regler som är beskrivna ovan för att svara på nedanstående frågor.

- Kan vi veta vilken sjukdom Peter har? Och Mary?
- Vem har influensa?
- Vilka symptom har Peter?
- Vem har diarré?
- Vem är trött?
- Finns det någon medicin som hjälper Peter?
- Har John och Mary några gemensamma symptom?

Faktan i den första listan kommer att bli en serie av fakta på formen  $\text{har}(\text{peter}, \text{influensa})$ . Den andra listan blir en serie regler (“rules” i tutorial-dokumentet). Fakta och regler skrivs i programfilen. Den tredje listan blir frågor ställda till Prolog i prompten efter att filen laddats. Dessa frågor, samt de resultat som kommer utav dem, ska även finnas med i inlämningsfilen som kommentarer, längst ner.

Fundera även över vad du skulle behöva lägga till för fakta i databasen för att beskriva en ny sjukdom. Detta behöver dock inte finnas med i den inlämnade filen.