

Dat2/F6S/SW4: Syntaks og semantik

Manual

Uli Fahrenberg

Foråret 2008

Indhold

Indhold	1
1 Om manualen	1
2 Om kursets indhold	1
2.1 Hvilke emner rummer kurset?	1
2.2 Hvad kan det bruges til?	1
2.3 Hvordan hænger kursets indhold sammen med kurset ‘Sprog og oversættere’?	2
2.4 Hvad er målene for kurset?	2
3 Om kursets form	3
3.1 Hvordan er en kursusgang opbygget?	3
3.2 Hvad forberedelse forventes der til kursusgangene?	3
3.3 Hvordan forløber opgaveregningen?	3
3.4 Er der afleveringsopgaver?	3
4 Kursusmateriale	3
4.1 Hvilken bog skal jeg bruge?	3
4.2 Hvad er kursets hjemmeside?	4
4.3 Er der en kursusplan?	4
5 Om eksamen	5
5.1 Hvordan bliver eksamen?	5
5.2 Hvad skal jeg kunne til eksamen?	5
5.3 FAQ	7

1 Om manualen

Dette er manualen for kurset ‘Syntaks og semantik.’ På de følgende sider vil jeg prøve at besvare alle de vigtige og typiske spørgsmål, du måtte have som studerende. **Læs denne manual grundigt igennem ved kursets start og læs i den igen, hvis der er noget du kommer i tvivl om.**

Manualen følger kurset, så denne manual er en opdateret udgave af manualen fra foråret 2007, som i sig selv var en opdateret udgave af Hans Hüttels manual fra foråret 2006. Forfatteren takker Hans for det store arbejde han har lagt i manualen og i kurset generelt.

2 Om kursets indhold

2.1 Hvilke emner rummer kurset?

I dette kursus stifter vi bekendtskab med grundlæggende matematiske modeller for programmeringssprogs syntaks og semantik. Teorien om programmeringssprog er et centralt område af datalogi. Det er vigtigt, når man skal designe et nyt programmeringssprog, lære et nyt programmeringssprog, skrive en compiler eller fortolker eller beskæftige sig med verifikation af programmer.

Kurset falder i to hoveddele, med ca. 7 minimoduler til hver:

- *Syntaksdelen* af kurset handler om hvordan man kan beskrive et programmeringssprogs *form* – hvornår er en programtekst ‘fejlfri’ og hvordan kan man finde ud af om en programtekst er ‘fejlfri’?
- *Semantikdelen* handler om programmeringssprogs *adfærd* – hvordan kan man beskrive udførelsen af et program uden at skulle tale om en konkret implementation med alle dens krummerlyrer og maskinafhængigheder?

Den opmærksomme læser vil have bemærket at $15 - (7 + 7) = 1$, dvs. der er ca. én overskydende minimodul. Denne sidste kursusgang skal vi bruge på at opnå en dybere teoretiske forståelse for et af de helt centrale emner i kurset, nemlig *rekursion og fikspunkter*.

2.2 Hvad kan det bruges til?

Kursets indhold kan umiddelbart anvendes til f.eks. beskrivelse og design af programmeringssprog, se næste afsnit. Det lægger også grobund for kurset ‘Kompleksitet og beregnelighed’ på næste semester, der beskæftiger sig med hvad det egentlig er computere kan beregne (og hvad de *ikke* kan).

På et andet plan vil den *matematiske modenhed* I (forhåbentlig) opnår gennem dette kursus være jer til gavn under hele jeres resterende uddannelse.

2.3 Hvordan hænger kursets indhold sammen med kurset ‘Sprog og oversættere’?

Programmeringssprogs form og adfærd er også en del af kurset ‘Sprog og oversættere’ som undervises af Bent Thomsen. Men hvor vores kursus fokuserer på teoretiske aspekter ved *beskrivelse* af programmeringssprog, handler Bents kursus om *design* af programmeringssprog og konstruktion af oversættere. Vores kursus udgør således et vigtigt teoretisk fundament for Bents kursus, og omvendt kan Bents kursus give konkrete eksempler på ting fra vores kursus.

Desuden er de abstraktioner vi vil snakke om nyttige i en del sammenhænge udenfor oversætterkonstruktion, f.eks. i områder som programverifikation eller genkendelse af naturlige sprog.

2.4 Hvad er målene for kurset?

Et vigtigt mål med kurset er selvfølgelig at I skal forstå og kunne anvende de begreber og værktøjer der bliver introduceret. Men det er ikke det eneste: Et ligeså vigtigt formål er at I trænes i *matematisk modenhed*.

Matematisk modenhed er et vidt begreb som er svært at definere. I forhold til vores kursus drejer det sig mest om det der kaldes henholdsvis *symbolkompetence* og *ræsonnementskompetence*: I trænes i at læse og skrive matematisk symbolsprog og i at relatere det til naturligt sprog, og i at læse og skrive matematiske argumenter, specielt beviser. Begge kompetencer er vigtige indenfor datalogi, specielt når man beskæftiger sig med mere grundlæggende aspekter af datalogiske problemstillinger.

Mere præcist er målene for kurset at I

- lærer at formidle jer mundtligt inden for kursets emner,
- lærer at formidle jer skriftligt indenfor kursets emner, herunder at I bliver fortrolige med anvendelsen af matematisk notation,
- lærer at formidle jer præcist,
- får et *aktivt* forhold til kursets emner,
- får overblik over kursets emner og hvordan de hænger sammen,
- opnår et præcist kendskab til alle centrale begreber i kurset,
- opnår et præcist kendskab til områdets centrale resultater, herunder hvordan man beviser dem,
- lærer at relatere de centrale begreber og resultater til hinanden, og
- får kompetence i brug af matematisk modellering som redskab i sprogdesign og sprogimplementation.

3 Om kursets form

3.1 Hvordan er en kursusgang opbygget?

En kursusgang i ‘Syntaks og semantik’ kommer til at forløbe således:

8:15–10:00 Forelæsning

10.10–12:00 Opgaveregning

3.2 Hvad forberedelse forventes der til kursusgangene?

Jeg forventer at I hurtigt skimter forelæsningsens stof *inden* forelæsningsen, og læser det grundigt igennem efter kursusgangen. Jeg vil en gang imellem bruge noget af forelæsningsstunden til “spørgetime,” hvor I forventes at stille spørgsmål til bøgernes eller min fremstilling af stoffet.

3.3 Hvordan forløber opgaveregningen?

Opgaveregningen er den vigtigste del af undervisningen. Regn opgaverne i fællesskab; det har vist sig at være helt klart den bedste måde. Bestem evt. en ord- og kridtfører.

Grupperummene er undervisningslokaler. Studerende, der *ikke* deltager i opgaveregningen, kan derfor ikke opholde sig i grupperummene under opgaveregningen.

3.4 Er der afleveringsopgaver?

Jeg vil stille to større opgaver i løbet af kurset, som I forventes at bruge en del af hver kursusgangs opgaveregningstid på. Disse skal afleveres *gruppevist* og vil hver danne grundlag for et af eksamensspørgsmålene.

De studerende der følger kurset som PE-kursus kan vælge at erstatte en eller begge opgave(r) med en der relaterer sig direkte til deres projekt.

4 Kursusmateriale

4.1 Hvilken bog skal jeg bruge?

I *syntaksdelen* benytter vi os af følgende bog:

Michael Sipser: *Introduction to the Theory of Computation*, Second Edition, PWS Publishing Co. 2005.

Der findes en ældre udgave af bogen. Lad være med at købe den – den er forældet. Alt materiale i dette kursus henviser til den nye udgave.

I *semantikdelen* bruger vi en note af min kollega Hans Hüttel:

Hans Hüttel: *Pilen ved træets rod*, Aalborg Universitet 2007.

Den vil kunne købes i boghandelen i løbet af februar. (Vi skal jo ikke bruge den før ottende kursusgang.)

4.2 Hvad er kursets hjemmeside?

Den er her:

<http://sands08.twoday.net>

Jeg har for vane at bruge blogs som kursushjemmesider, da det

1. gør livet nemmere for mig og
2. giver *jer* mulighed for at *kommentere* kurset. Gør endelig brug af den mulighed; det er en god måde at feedbacke (findes det ord?) på.

4.3 Er der en kursusplan?

Sidste år så den ud som nedenfor, og jeg regner ikke med at vi laver meget om:

Syntax – regulære sprog

1	Introduktion; sprog; regulære udtryk	Sipser afs. 0.2, 1.3
2	Endelige automater	Sipser afs. 1.1
3	Nondeterministiske automater	Sipser afs. 1.2
4	Sprog der <i>ikke</i> er regulære	Sipser afs. 1.4

Syntax – kontekstfrie sprog

5	Kontekstfrie grammatikker	Sipser afs. 2.1
6	Pushdown-automater	Sipser afs. 2.2
7	Sprog der <i>ikke</i> er kontekstfrie	Sipser afs. 2.3

Semantik

8	Introduktion til (operationel) semantik	HH, kap. 1 og 3
9	Operationelle semantikker for Bims	HH, kap. 4
10	Udvidelser af Bims	HH, kap. 5
11	Blokke og procedurer	HH, kap. 6
12	Parametermekanismer	HH, kap. 7
13	Semantikopgaven	

Teoretisk grundlag

14	Rekursive definitioner	HH, kap. 14
15	Denotationel semantik	HH, kap. 13

5 Om eksamen

5.1 Hvordan bliver eksamen?

Eksamen er mundtlig og varer 20 minutter per studerende. Der er et antal eksamensspørgsmål som I kender på forhånd. Dem trækker I et af, hvorefter I får 20 minutter til at forberede en besvarelse af spørgsmålet der forventes at vare 10 til 15 minutter. Eksamenspræstationen bedømmes af en *ekstern* censor, og der gives karakter.

Eksamenspensum og -spørgsmålene vil blive fastlagt ved tredjesidste kursusgang. Syntaks- og semantikopgaverne er del af eksamenspensum.

5.2 Hvad skal jeg kunne til eksamen?

Når censor og jeg bedømmer en eksamenspræstation, forsøger vi at besvare en række spørgsmål:

Har den studerende overblik? Her ser vi efter i hvor høj grad I kan gøre rede for

- hvilke begreber der er væsentlige.

- hvorfor det netop er disse begreber, der er de væsentlige.
- hvilke resultater der er væsentlige. I et kursus som *Syntaks og semantik* er de vigtige resultater normalt *sætninger*.
- hvorfor det netop er disse resultater, der er de væsentlige.
- hvilke sammenhænge der er mellem denne del af stoffet og andre dele af stoffet.

Kan den studerende formulere sig præcist? Her ser vi efter i hvor høj grad I kan gøre rede for

- hvordan de vigtige begreber er defineret. I et kursus som *Syntaks og semantik* bliver de vigtige begreber indført ved præcise *definitioner*.
- hvad de forskellige størrelser, der indgår i en definition, betegner.
- hvor og hvordan vi bruger definitionerne.
- hvorfor definitionerne er formuleret som de er.
- hvordan de vigtige sætninger er formuleret, og hvilke antagelser der er gjort i sætningernes formulering.

Kan den studerende tænke om emnet og forstå dets tankegange? Her ser vi efter i hvor høj grad I kan gøre rede for

- strukturen i et ræsonnement, hvad de væsentlige ideer er og hvorfor de er væsentlige. I et kursus som *Syntaks og semantik* er der især vigtige ræsonnementer i *beviser* for sætninger.
- hvorfor vi får bevist vores sætning ved at foretage netop disse bevisskridt.
- hvor antagelser og definitioner bliver brugt i et ræsonnement.
- hvilke beviste knikker der er vigtige i det aktuelle emne.
- hvilke konsekvenser en sætning har.

Er den studerende selvstændig? Her ser vi efter i hvor høj grad I kan formulere jer selvstændige. Vi ser efter

- i hvor høj grad I kan tale om emnet uden at læse højt eller skrive af fra dispositionen.
- hvad det er, I bruger dispositionen til. (Er det til at hjælpe med struktur? Er det for at få styr på detaljer? Skal alle de vigtige begreber læses op/skrives af?)
- hvordan I reagerer, hvis I bliver præsenteret for en problemstilling (f.eks. et eksempel), der ikke er kendt på forhånd.

5.3 FAQ

**Hvorfor skal vi lære om alle de små opfundne legetøjssprog?
Hvorfor kan vi ikke se på hele Java eller C eller ...?**

Semantikdelen af kurset handler om hvordan man beskriver forskellige fænomener set isoleret, og derfor er det godt at vælge simple sprog der kun rummer disse fænomener. Senere skal I lære at betragte fænomenernes samspil i stadig mere komplekse sammenhænge. En analogi: I et indledende kursus i mekanisk fysik lægger man heller ikke ud med at beskrive komplicerede mekaniske systemer med gnidningsmodstand, tyngdekraft osv. inddraget på samme tid.

Hvorfor er der alle de beviser?

I et universitetsstudium skal man bl.a. erhverve sig en teoretisk indsigt i studiets områder, og dette indebærer at man skal kende til teoriområdernes argumentation. Angrebsvinklen i dette kursus er primært teoretisk, og de teorier, vi skal se på, er matematisk funderede teorier. Resultaterne i matematisk orienterede discipliner er matematiske *sætninger*, og argumentationsformen er matematiske *beviser*. Så derfor er det denne argumentationsform, du vil få at se. Andre discipliner i datalogi benytter sig af andre argumentationsformer.

Samtidig er der også en læringsbetinget grund til at gøre beviserne for kursets sætninger til genstand for en mere nøje betragtning. Matematiske sætninger siger nemlig typisk noget om, *hvordan begreber hænger sammen*, og man kan derfor lære at relatere kursets begreber til hinanden ved at betragte beviserne. Et eksempel er resultatet om sammenhængen mellem pushdown-automater og kontekstfrie grammatikker (Sipser Lemma 2.15): Beviset fortæller præcis hvordan man kan konstruere en grammatik ud fra en automat, og ved at I sætter jer ind i beviset, får I forhåbentlig en bedre forståelse af både grammatikker og automater.