

*Syntaks og semantik*  
Svar på tekstspørgsmål

**Rasmus Hoppe Nesgaard Aaen**

Tirsdag 26. april 2011, 09:23:17

## Kursusgang 2

1. En NFA og en DFA er begge defineret som 5-tupler. Hvad er de præcise forskelle på en NFA og en DFA?

**Svar:** NFA = Nondeterministisk Endelig Automat, DFA = deterministisk Endelig Automat

De fungerer på samme måde, men den nondeterministiske endelige automat, kan have flere set tilstande til et input, fx. kan „a“ fra en enkelt tilstand godt have 0 eller flere tilstande at gå til.

2. Overføringsfunktionen  $\delta$  kan beskrives med en tabel. Hvis en NFA ikke har nogen transitioner mærket  $b$  fra en tilstand  $q$ , hvordan kommer det da til udtryk i tabellen for  $\delta$ ?

**Svar:** Vi markerer dette ved at skrive „Ø“.

3. Forklar med egne ord og dernæst med en præcis definition, hvornår en NFA  $M$  accepterer en inputstreng  $w$ .

**Svar:** Med egne ord: En NFA  $M$  acceptere en inputstreng  $w$ , hvis en af „kopierne“ af  $M$  er en accept tilstand med inputstrengen  $w$ .

4. Hvad er det vigtige resultat om sammenhængen mellem NFA'er og DFA'er?

**Svar:** Hvis et sprog er genkendt af en NFA, kan det også genkendes af en tilsvarende DFA.

5. Hvis  $S$  er en mængde af tilstande, hvad betegner  $E(S)$  da? Og hvorfor indfører bogen i det hele taget denne notation?

**Svar:**  $E(S)$  er mængden af tilstande vi kan nå fra  $S$  med 0 eller flere  $\epsilon$

6. Findes der en systematisk metode til at konvertere en NFA til en DFA, eller er man nødt til at prøve sig frem? Begrund dit svar så præcist som muligt og forklar grundideer i metoden, hvis den findes.

**Svar:**

7. Er de regulære sprog lukket under  $*$ ? Hvis ja, forklar præcist hvorfor. Hvis nej, forklar hvorfor ikke.

**Svar:**

## Kursusgang 3

Intet bidrag endnu.

## Kursusgang 4

Intet bidrag endnu.

## Kursusgang 5

1. I dagens tekst ser vi to forskellige pile, nemlig  $\Rightarrow$  og  $\rightarrow$ . Hvad er forskellen på dem?

**Svar:**  $\Rightarrow$  er Derivationer.  
 $\rightarrow$  er regler.

2. Hvad er forskellen på et kontekstfrit sprog og en kontekstfri grammatik?  
Giv en præcis, kort forklaring.

**Svar:**

3. I kladden til en lærebog har forfatteren skrevet dette:

Sproget defineret af en kontekstfri grammatik  $G = (V, \Sigma, R, S)$   
er givet ved

$$L(G) = \{w \mid S \Rightarrow w\}$$

Forklar, hvorfor dette ikke er korrekt og skriv det rigtige svar.

**Svar:**

4. En kontekstfri grammatik kan være tvetydig. Hvordan er dette begreb defineret?

**Svar:**

5. Hvad siger det vigtige resultat om Chomsky-normalform i dagens tekst?

**Svar:**

6. Gælder det, at en grammatik på Chomsky-normalform ikke er tvetydig?  
Hvis ja, forklar hvorfor. Hvis nej, find et meget lille modeksempel.

**Svar:**

## Kursusgang 6

1. Hvordan kan vi se af definitionen af en pushdownautomat, at den kan være nondeterministisk?

**Svar:** Fordi den tomme streng kan indgå i Alfabetet  $\Sigma_\epsilon$  vi bruger til overførsels tilstanden og fordi overførsels tilstanden kan have flere tilstande.

2. Hvordan kan vi se af definitionen af en pushdownautomat, at automaten kan tilføje et element til stakken (*pushe*) ?

**Svar:** Det ses i overføringsfunktionen  $\Gamma$

3. Hvordan kan vi se af definitionen af en pushdownautomat, at automaten kan fjerne topelementet af stakken (*poppe*) ?

**Svar:** Det ses i overføringsfunktionen

4. Præcis hvad har pushdownautomater med kontekstfrie sprog at gøre?

**Svar:** Et sprog er kontekstfrit hvis en PDA genkender det.

5. I dagens tekst indfører vi nogle variabler, vi kalder  $A_{pq}$ . I hvilken sammenhæng optræder de, og hvad er deres rolle?

**Svar:**  $A_{pq}$  bruges i forbindelse med beviset for at generering af CFG fra PDA'er.

6. Gælder det, at man kan konvertere en nondeterministisk pushdownautomat til en deterministisk? Hvis ja, forklar hvor dette er beskrevet i dagens tekst og opsummer argumentet. Hvis nej, forklar hvor dette er beskrevet i dagens tekst og opsummer argumentet.

**Svar:** Side 111: „Nondeterministic pushdown automata recognize certain languages which no deterministic pushdown automata can recognize, though we will not prove this fact.“

7. Findes der regulære sprog, der ikke er kontekstfrie? Hvis ja, så forklar præcis hvorfor. Hvis ikke, så forklar præcis hvorfor.

**Svar:** Alle regulære sprog er kontekstfrie fordi en endelig automat er en PDA bare uden stack, se side 122 i bogen.

## Kursusgang 7

1. Kan Pumping Lemma bruges til at bevise, at et sprog er kontekstfrit? Hvis ja, så forklar hvordan man bærer sig ad. Hvis nej, så forklar hvorfor dette ikke er muligt.

**Svar:** Nej, fordi vi normalt bruger Pumping Lemma til at bevise at et sprog ikke er kontekstfrit.

2. Der dukker et  $b$  op i beviset for Pumping Lemma. Hvad betegner  $b$ , og hvorfor er det vigtigt?

**Svar:**  $b$  is the maximum number of symbols in the right-hand side of a rule. A node can have no more than  $b$  children. At most  $b^h$  leaves are within  $h$  steps of the start variable.

3. I beviset for Pumping Lemma vælges  $p$  til  $p = b^{|V|} + 1$ . Hvad er  $|V|$  og hvorfor er det en god idé at vælge  $p$  til denne værdi?

**Svar:**

4. I beviset for Pumping Lemma vælger vi det parsetræ for strengen  $s$ , som har færrest knuder. Forklar, hvor i beviset dette bliver vigtigt.

**Svar:**

5. I beviset for Pumping Lemma betragter vi den *nederste* gentagelse af en variabel  $R$ . Forklar, hvor i beviset dette bliver vigtigt.

**Svar:**

6. Her er et forsøg på at bevise at sproget  $L_1 = \{ww \mid w \in \{a,b\}^*\}$  ikke er kontekstfrit. Hvis beviset er korrekt, så forklart hvorfor. Hvis beviset er forkert, så forklar, præcis hvad der er galt med det.

Hvis  $L_1$  er kontekstfrit, er der en  $p > 0$  og en  $s \in L_1$  så  $s$  kan opsplittes, så betingelserne 1-3 i Pumping Lemma er overholdt. Men vi kan vælge  $s = a^p b^p a^p b^p$ . Det er klart at  $s \in L_1$ . Opsplitningen  $u = \varepsilon$ ,  $v = a$ ,  $x = a^{p-2}$ ,  $y = a$  og  $z = b^p a^p b^p$  overholder betingelserne 2 og 3, men  $uv^2xy^2z \notin L_1$ . Derfor er  $L_1$  ikke kontekstfrit.

**Svar:**

## Kursusgang 8

1.  $xRy$  skal læses som ‘ $x$  elsker  $y$ ’. Forklar på almindeligt dansk, hvordan nedenstående logiske udsagn, der bruger kvantorer, skal læses.

(a)  $\exists x.xRx$

**Svar:** Der findes mindst én  $x$  hvor  $x$  elsker  $x$ .

(b)  $\forall x.\exists y.xRy$

**Svar:** For alle  $x$  findes der mindst én  $y$  hvor  $x$  elsker  $y$ .

(c)  $\forall x.\forall y.xRy$

**Svar:** For alle  $x$  og for alle  $y$ , findes der en  $x$  der elsker  $y$ .

2. Lad  $A$  og  $B$  være mængder. Hvad betyder notationen  $A \rightarrow B$  helt præcist? (‘ $A$  går over i  $B$ ’ er et meget dårligt svar.)

**Svar:**  $A$  er domænet af en funktion,  $B$  er rækkevidden.

3. Lad  $A$  og  $B$  være mængder. Hvad betyder notationen  $A \rightharpoonup B$  helt præcist? (‘ $A$  går over i  $B$ ’ er et meget dårligt svar.)

**Svar:**  $A$  er domænet af en partiel funktion,  $B$  er rækkevidden.

4. Herunder er tre sprog-elementer fra **Bims**. Angiv for hvert af dem hvilken *syntaktisk kategori* det tilhører og hvad dets *umiddelbare bestanddele* er:

- $\underline{7} + (\underline{3} * \underline{5})$

**Svar:** *Syntaktisk kategori* Aritmetisk udtryk, *umiddelbare bestanddele*  $\underline{7}$  og  $(\underline{3} * \underline{5})$ .

- `while 0 = 0 do skip`

**Svar:** *Syntaktisk kategori* Kommando, *umiddelbare bestanddele* 0 og 0.

- `x := 4; y := 2`

**Svar:** *Syntaktisk kategori* Kommando, *umiddelbare bestanddele* 4 og 2.

5. Angiv de forskellige komponenter i et transitionssystem – hvordan de noteres og hvad terminologien for dem er.

**Svar:**  $\Gamma$  er mængden af konfigurationer,  $\rightarrow$  er transitions relationen,  $T$  er slut konfigurationer.

6. Er transitionsreglen

$$\frac{a_2 + a_1 \rightarrow v}{a_1 + a_2 \rightarrow v}$$

kompositionel? Hvis ja, så forklar hvorfor. Hvis nej, så forklar hvorfor ikke.

**Svar:**

7. Er transitionsreglen

$$\frac{a_1 \rightarrow v_1 \quad a_2 \rightarrow v_2}{a_1 + a_2 \rightarrow v} \quad \text{hvor} \quad v = v_1 + v_2$$

kompositionel? Hvis ja, så forklar hvorfor. Hvis nej, så forklar hvorfor ikke.

**Svar:**

## Kursusgang 9

1. En studerende blev til en eksamen spurgt, hvad en programtilstand er i bogens kapitel 4. Her er, hvad han svarede:

*En tilstand er et øjeblikkeligt syn på vores program; der hvor det står. Det er alt hvad vi ved om programmet, skridt, linjenumre osv.*

Hvad er det rigtige, præcise svar ifølge bogen? (Det er en dårlig idé ikke at bruge bogens notation.)

**Svar:**

2. I dagens tekst optræder både  $S$  og  $s$ . Betegner de det samme? Hvad betegner de egentlig?

**Svar:**  $S$  er en kommando,  $s$  er en tilstand.

3. Er alle reglerne i big-step-semantikken for **Bims** kompositionelle? Hvis ja, hvorfor? Hvis nej, hvilke regler er da ikke kompositionelle og hvorfor ikke?

**Svar:** Nej, while er ikke kompositionel, fordi den bliver til både while true og while false.

4. Hvordan kan vi af big-step-reglerne for **while**  $b$  **do**  $S$  se, at betingelsen  $b$  skal evalueres *inden* løkkens krop kan blive udført?

**Svar:** Man skal evaluere  $b$  før man kan finde ud af hvilken metode man skal bruge på kommandoen.

5. Hvor mange regler er der for **while**  $b$  **do**  $S$  i *small-step-semantikken*?

**Svar:** Der er 1.

6. Hvordan kan vi af small-step-reglerne for  $S_1; S_2$  se, at  $S_1$  skal udføres inden  $S_2$ ?

**Svar:**  $S_2$  afhænger af evalueringen af  $S_1$

7. Hvad er det vigtigste *resultat* i dagens tekst ?

**Svar:**

8. Hvorfor er induktion i transitionsfølgers længde ikke noget særlig nyttigt bevisprincip for vor big-step-semantik?

**Svar:** Fordi man er nødt til at udregne uendelige løkker til enden, hvilket virker rimeligt umuligt.

## Kursusgang 10

1. Hvordan kan man af big-step-reglerne for repeat-løkker se, at betingelsen skal evalueres *efter* hvert gennemløb af løkkens krop? Vær så præcis som muligt; brug bogens notation.

**Svar:** Den skal udføre kommandoen  $S$  indtil betingelsen  $b$  evaluerer til sand. Det ses ud fra reglerne, at udtrykket, uanset om det er [REPEAT-FALSK] eller [REPEAT-SAND], resulterer i  $s \rightarrow s'$ . (se side 64)

2. Hvad har Sætning 5.2 med semantisk ækvivalens at gøre? Vær så præcis som muligt i dit svar; brug bogens notation.



**Svar:** Sætningen siger, at  $\langle \text{repeat } S \text{ until } b, s \rangle \rightarrow s'$  gælder, hvis  $\langle S; \text{while } \neg b \text{ do } S, s \rangle \rightarrow s'$  gælder. Dvs. de to udtryk er semantisk ækvivalente når de, givet den samme start-tilstand, returnerer den samme slut-tilstand.

3. I beviset for Sætning 5.2 benytter vi, at  $s \vdash b \rightarrow \#$  hvis og kun hvis  $s \vdash b \rightarrow ff$ . Hvordan kan vi se, at dette gælder? Giv et præcist svar.

**Svar:**

4. En studerende blev til eksamen bedt om at definere big-step semantisk ækvivalens. Her er hvad han sagde:

$S_1$  og  $S_2$  er big-step semantisk ækvivalente, hvis der findes en  $s$  så  $\langle S_1, s \rangle \rightarrow s'$  og  $\langle S_2, s \rangle \rightarrow s'$

Er dette et korrekt svar? Hvis ja, så forklar hvorfor. Hvis nej, så giv det rigtige og præcise svar. Vær så præcis som muligt; brug bogens notation.

**Svar:** Ja, det er korrekt. jvf bogen side 67, så er to kommandoer  $S_1$  og  $S_2$  semantisk ækvivalente hvis de, givet den samme starttilstand, returnerer den samme slutttilstand.

5. En anden studerende blev til eksamen bedt om at definere big-step semantisk ækvivalens. Her er hvad han sagde:

Big-step semantisk ækvivalens er defineret ved at

- $S_1 \sim_{bss} S_1$ ,
- hvis  $S_1 \sim_{bss} S_2$  så også  $S_2 \sim_{bss} S_1$ , og
- hvis  $S_1 \sim_{bss} S_2$  så også  $S_2 \sim_{bss} S_3$ , så også  $S_1 \sim_{bss} S_3$ , og

Er dette definitionen af semantisk ækvivalens? Hvis ja, så forklar hvorfor. Hvis nej, så forklar hvorfor ikke, og hvad det så er, han har prøvet at udtrykke. Vær så præcis som muligt; brug bogens notation.

**Svar:**

Nej, det ser ikke ud til at være den korrekte definition på semantisk ækvivalens. Man kan ikke starte med at sige at  $S_1 \sim_{bss} S_1$ , eller det kan man godt, men det giver ingen mening at sige at en kommando er semantisk ækvivalent med sig selv, hvis  $S_1 \sim_{bss} S_2$  så også  $S_2 \sim_{bss} S_1$ , osv.

6. Hvor mange transitionsregler er der for **abort**-kommandoen i big-step-semantikken? Og i small-step-semantikken? Hvorfor?

**Svar:** Der er ingen transitionsregler for **abort** i nogen af dem, da der pr. definition kun er en transition, hvis vi kan finde et derivationstræ for den, ved brug af vores transitionsregler.

7. Hvorfor gælder det at **while 0=0 do skip**  $\sim_{ss}$  **abort**? Vær så præcis som muligt; brug bogens notation.

**Svar:** Der er semantisk ækvivalens mellem uendelige løkker og abnorm terminering.

8. Hvad er forskellen på god (*angelic*) og ond (*demonic*) nondeterminisme? Vær så præcis som muligt; brug bogens notation.

**Svar:** Big-step-semantik undertrykker uendelige løkker, dvs. uheldige valg giver ikke anledning til nogen transitioner. Dette kan man kalde *god nondeterminisme*.

Modsat er det i small-step-semantik, hvor uendelige løkker ikke undertrykkes, dvs. der dannes en uendelig transitionsfølge hvis man træffer et uheldigt valg (s. 72-73).

## Kursusgang 11

1. Hvad betegner  $env_V$  i dagens tekst ?

**Svar:** Et vilkårligt element i **EnvV**.

2. Hvad betegner **EnvV** i dagens tekst ?

**Svar:** Dette betegner mængden af variabelenvironments som er mængden af partielle funktioner fra variabler til lokationer.

3. I dagens tekst optræder både next og new. Hvad er de, og hvordan er de defineret?

**Svar:** next giver den næste ledige lokation. Hvis man kalder new til en lokation, så returneres den næste lokation, fri eller ej.

4. Forklar ved brug af environment-store-modellen indholdet af transitionsreglen

---


$$[\text{var-bip}_{\text{bss}}] \quad env_V, sto \vdash x \rightarrow_a v \quad \text{hvis} \quad env_V x = l \text{ og } sto l = v$$


---

**Svar:** –

5. Præcis eet af nedenstående udsagn om big-step-semantikkerne i dagens tekst er korrekt. Hvilket er korrekt og hvorfor? Forklar, hvorfor netop dette udsagn bør være sandt for **Bip**.

- Udførelse af en kommando  $S$  kan ændre både variabel-environment og store
- Udførelse af en kommando  $S$  kan ændre variabel-environment
- Udførelse af en kommando  $S$  kan ændre store

**Svar:** Udførelse af en kommando  $S$  kan ændre store (se s. 84).

6. Definer **EnvP** i tilfældet hvor vi antager dynamiske scoperegler for variable og statiske scoperegler for procedurer. Forklar, hvordan vi af definitionen af **EnvP** kan se, at der er tale om netop disse scoperegler.

**Svar:** **EnvP** giver udførelsen af en procedure  $p$  med variabelbindingerne på kaldtidspunktet, men med procedurebindingerne på  $p$ 's erklæringstidspunkt.

7. Her er en big-step-transitionsregel. Forklar hvilken kombination af scoperegler den udtrykker.

---


$$(\text{CALL-1}) \quad \frac{env_V, env'_P \vdash \langle S, sto \rangle \rightarrow sto'}{env_V, env_P \vdash \langle \text{call } p, sto \rangle \rightarrow sto'}$$

hvor  $env_P p = (S, env'_P)$

---

**Svar:** Den udtrykker *dynamisk binding af variabler og statisk binding af procedurer*.

8. Betragt tilfældet, hvor vi antager dynamiske scoperegler for variabler og procedurer, og antag at der i en procedure  $p$  med krop  $S$  ikke også findes en lokal procedure med samme navn. Forklar, hvorfor alle kald af  $p$  i kroppen  $S$  da vil være rekursive.

**Svar:** –

## Kursusgang 12

Intet bidrag endnu.

## Kursusgang 13

1. Hvad betyder det, at parallelle komponenter i sproget i afsnit 8.1 skal være *variabel-uafhængige*? Og hvorfor er det engod idé at have denne antagelse?

**Svar:** De har ingen variabler til fælles, og kan derfor kører parallelt uden at have nogen effekt på hinanden.

2. Hvad skal kommandoerne  $c?x$  og  $c!a$  bruges til?

**Svar:**  $c?x$  modtager  $v$  og binder til  $x$ ,  $c!a$  sender  $a$  på kanal  $c$ .

3. Hvorfor indfører vi mærkede transitionssystemer? Hvordan er de defineret? Og hvad er forskellen fra de transitionssystemer, vi tidligere har set?

**Svar:** Grunden til at vi bruger mærkede transitioner er at vi ikke altid kan garantere at modtageren modtager netop det der blev sendt. og defineret:  $(\Gamma, A, \rightarrow a)$   
 $\Gamma = Stm \ x \ states$

4. Hvad er forskellen på asynkron og synkron kommunikation?

**Svar:** I asynkron kommunikation bliver en sendt besked gemt i et medium, hvor en eventuel modtager senere kan hente den. Synkron kommunikation går selvfølgelig ud på det modsatte, nemlig at der modtages i det øjeblik der sendes.

5. Hvorfor er det nødvendigt at skrive `remove(c, m)` i konfigurationen på højre side af reglen  $INPUT_{L-SSS}$ ?

**Svar:** Den fjerner værdien fra mediet, efter den er blevet hentet af modtageren. Det er nødvendigt for at sikre at modtageren ikke henter den samme værdi to gange.

6. Hvor er begrebet en *ækvivalensrelation* defineret i bogen? (Det er ikke i dagens tekst!) Hvordan ser definitionen ud?

**Svar:**

## Kursusgang 14

1. Hvad er basistyperne i dagens tekst?

**Svar:** int og bool.

2. Hvilken type har et procedurenavn  $p$ ?

**Svar:** Bool.

3. Hvad er det, et typeenvironment indeholder oplysninger om? Hvordan er det defineret?

**Svar:** Fortæller hvad typerne af variabler og procedurenavne er.

4. Er der typeregler, der direkte gør brug af typeenvironmentet? Hvis ja, hvad hedder de da?

**Svar:** DecV, Decp, Exp, stm.

5. I dagens tekst dukker notationen

$$env_V, env_P \vdash_E \langle S, sto \rangle \rightarrow \text{wrong}$$

op. Forklar, hvad den betyder.

**Svar:**

6. Hvad er det væsentligste resultat i dagens tekst, og hvad siger det helt præcist? (Og vær nu helt præcis, tak.)

**Svar:**

7. Hvad betegner funktionen  $\text{set}(T)$ ?

**Svar:**

8. Hvad menes der med at et typesystem har *slack*?

**Svar:**

## Kursusgang 15

1. Forklar på ikke under 5 linjer og ikke over 10 linjer, hvad fikspunkter og rekursion har med hinanden at gøre.

**Svar:**

2. Giv tre forskellige eksempler på partielle ordninger (po'er). Højst ét af dem må have med tal at gøre.

**Svar:**

3. En studerende blev ved eksamen bedt om at definere, hvad en cpo er. Her er, hvad han svarede:

*En cpo er det samme som en voksende følge, og den har en grænseværdi.*

Var det et godt svar? Hvorfor ikke? Hvad burde han have svaret?

**Svar:** Måske

4. Er mængden af naturlige tal  $\mathbb{N}$  ordnet mht.  $\leq$  en cpo? Hvorfor/hvorfor ikke?

**Svar:**

5. En anden studerende blev ved eksamen bedt om at definere, hvad en kontinuert funktion er. Her er, hvad han svarede:

*En kontinuert funktion er en, der ligesom er helt glat. Den har ingen huller i sin graf.*

Var det et godt svar? Hvorfor ikke? Hvad burde han have svaret?

**Svar:** Måske.

6. Hvad er et fikspunkt?

**Svar:**

7. Giv et simpelt eksempel på en funktion, der har et fikspunkt, og på et fikspunkt for denne funktion.

**Svar:**

8. Hvad er det vigtigste resultat i dagens tekst? Hvorfor er det vigtigt?

**Svar:**