

DD1351 - DD1350 Logik för dataloger

HEMTAL FÖR BETYG D-A

Johan Karlander
KTH EECS

Hemtalet skall vara inskickat i canvas senast den 21 april kl 23.00. Alla uppgifter skall lösas individuellt. I övrigt är alla hjälpmedel tillåtna.

Del C

För att uppnå betyg C krävs 7 poäng, och för betyg D krävs 4 poäng på C-delen. I bägge fallen krävs även godkänt på E-delen.

1. Ge bevis i naturlig deduktion för följande:

4p

- (a) $\forall x (g(f(x)) = x), \forall y \exists x (f(x) = y) \vdash \forall y (f(g(y)) = y)$
- (b) $\forall y (f(g(y)) = y), \forall x \forall y (f(x) = f(y) \rightarrow x = y) \vdash \forall x (g(f(x)) = x)$

2. Din uppgift är att konstruera en modell i CTL. Du får utnyttja din egen kreativitet och komma med förslag på hur den ska se ut (med tillstånd och övergångar). Vi ger dock vissa specifikationer. Den ska ha ett starttillstånd s_1 . Sedan tänker vi oss att modellen beskriver en automat där vi kan skriva in symboler a, b, c . Det finns en hemlig kod. Om rätt kod skrivs in kommer man till tillstånd s_k . Inskrivandet av tecknen bör motsvaras av övergång mellan olika tillstånd. Från s_k ska det finnas en kant tillbaka till s_1 . I Vi bestämmer oss för att rätt kod är bac . Om man skriver in fel kod ska man få en ny chans att skriva in rätt kod. Det ska finnas en variabel *Start* som är sann i s_1 och ingenstans annars. Det ska också finnas en variabel *Processa* som är sann i s_k och ingenstans annars. I övrigt får du själv bestämma vilka variabler du vill ha. Konstruera en modell M sådan att $M, s_1 \models EF\text{ Processa}$ gäller. Avgör om $M, s_1 \models AF\text{ Processa}$ och $M, s_1 \models AX\text{ EF Start}$ också gäller i din modell.

3p

3. Vi ska (i denna uppgift) kalla en satslogisk formel för *tunn* om varje variabel förekommer högst en gång i formeln. Sådana tunna formler har vissa egenskaper som vi ska undersöka.

3p

- (a) Givet en sådan formel F , låt $V(F)$ vara antalet variabler i F och $K(F)$ vara antalet konnektiv i F . Vi antar att det är så att två negationer aldrig står bredvid varandra i F . Definiera $V(F)$ och $K(F)$ rekursivt.
- (b) Visa med strukturell induktion att det för varje tunn formel F gäller att $K(F) \leq 3V(F) - 2$.
- (c) Visa på lämpligt sätt att varje tunn formel är både satisfierbar och falsifierbar. (Så en tunn formel kan varken vara en tautologi eller en motsägelse.)

Del A

För att uppnå betyg A krävs 7 poäng, och för betyg B krävs 4 poäng på A-delen. I bägge fallen krävs även betyg C på C-delen.

4. I vanlig satslogik använder vi som bekant konnektiven $\neg, \vee, \wedge$ och $\rightarrow$. Vi vet också att man kan klara sig utan konnektivet $\rightarrow$. Vi kan skriva om varje formel av typen $A \rightarrow B$ på formen $\neg A \vee B$. Låt oss nu anta att vi arbetar i ett sådant system där implikation aldrig förekommer. (Genom att vi gjort lämpliga omskrivningar av formler.) Antag nu att vi också vill kunna bevisa saker i naturlig deduktion i det systemet. Men nu stöter vi på ett problem: Vi har ju de två härledningsreglerna $\rightarrow_e$ och $\rightarrow_i$ som inte kan användas längre. Ditt problem är nu att avgöra om vi kan klara oss utan reglerna eller om de kanske ska ersättas med nya regler. Kom ihåg att vi vill ha ett system som är sunt och fullständigt.

5p

5. Vi arbetar i Hoare-logik. Den här uppgiften är i två steg. Den första går ut på att du ska skriva ett program som testat om ett heltal delar ett annat heltal. Vi kallar programmet **DIV**. Mer speciellt ska programmet uppfylla följande specifikation:

$$\langle X = x_0 \wedge Y = y_0 \wedge 1 \leq x_0 \wedge x_0 \leq y_0 \rangle \text{DIV} \langle (Z = 1 \wedge \exists u(u \times x_0 = y_0)) \vee (Z = 0 \wedge \neg \exists u(u \times x_0 = y_0)) \rangle.$$

Skriv programmet och bevisa att programmet uppfyller partiel korrekthet givet för- och eftervillkoren.

Du ska nu konstruera ett program **PRIM** som avgör om ett heltal $N \geq 2$ är ett primtal eller inte. Du ska själv formulera lämpliga för- och eftervillkor. I programmet får du använda koden i programmet **DIV**. Bevisa att programmet uppfyller partiel korrekthet givet för- och eftervillkoren.
