

DD1350 Logik för dataloger

OBS: MODELLTENTA!

26 maj 2009, 14.00 - 16.00

Dilian Gurov

KTH CSC

Skriv ditt namn på alla blad. Formelblad kommer att delas ut under kontrollskrivningen; inga andra hjälpmedel är tillåtna. Kravet för godkänd på tentan är att vara godkänd på alla tre delar.

1 Satslogik, Predikatlogik

Kravet för godkänd på denna del är 8 poäng av 10. Om du har godkänd på första kontrollskrivningen, är du automatiskt godkänd på första uppgiften (och får alltså 5 poäng). Bonuspoäng från hemtalen 1 och 2 räknas tillgodo.

1. Låt ϕ vara predikatlogiska formeln

5p

$$\exists x (\forall y P(x, y, z) \wedge \exists z (P(x, y, z) \wedge \forall x P(x, y, z)))$$

och låt t vara termen $f(x, g(y))$. Med användning av pilar (eller olika färg), visa vilka kvantifikatorer som binder vilka variabelförekomster i formeln ϕ .

2. Presentera ett *bevis* i Naturlig deduktion till följande sekvent:

5p

$$p \vdash \forall x (Q(x) \rightarrow p)$$

Rita tydligt alla boxar för att visa räckvidden för alla antaganden och nya variabler i beviset.

2 Prolog, Induktion, Temporallogik

Kravet för godkänd på denna del är 8 poäng av 10. Om du har godkänd på andra kontrollskrivningen, är du automatiskt godkänd på första uppgiften (och får alltså 5 poäng). Bonuspoäng från hemtalen 3 och 4 räknas tillgodo.

1. Kom ihåg induktiva BNF-definitionen av listor som term mängder:

5p

$$List ::= \text{empty} \mid \text{cons} (Letter, List)$$

och induktiva definitionen av konkatenering **conc** (u, v):

$$\begin{aligned} \text{conc} (\text{empty}, v) &\stackrel{\text{def}}{=} v \\ \text{conc} (\text{cons} (a, u), v) &\stackrel{\text{def}}{=} \text{cons} (a, \text{conc} (u, v)) \end{aligned}$$

Definiera *induktivt* funktionen **reverse** (u) som vänder om på listor. Använd din definition för att beräkna stegvis:

$$\text{reverse} (\text{cons} (a, \text{cons} (b, \text{empty})))$$

Resultatet måste vara en korrekt lista!

2. Låt $Atoms \stackrel{\text{def}}{=} \{p, q\}$ och låt $\mathcal{M}$ vara modellen definierad som:

5p

$$\begin{aligned} S &\stackrel{\text{def}}{=} \{s_1, s_2\} \\ \rightarrow &\stackrel{\text{def}}{=} \{(s_1, s_2), (s_2, s_1), (s_2, s_2)\} \\ L &: \begin{aligned} s_1 &\mapsto \{p\} \\ s_2 &\mapsto \{q\} \end{aligned} \end{aligned}$$

Förklara intuitivt betydelsen av CTL formeln $\text{EG EX } p$ och avgör om $\mathcal{M}, s_1 \models \text{EG EX } p$ och $\mathcal{M}, s_2 \models \text{EG EX } p$ gäller. Motivera dina svar.

3 Hoare-logik

Kravet för godkänd på denna del är att ha löst en av de två E-uppgifterna. Bara då kommer C-uppgiften att kollas.

1. Betrakta följande program `Copy` som kopierar startvärdet av variabeln `x` till variabeln `y`:

E

```
a = x;
y = 0;
while (a != 0) {
    y = y + 1;
    a = a - 1;
}
```

Specificera programmet enligt total korrekthet med en Hoare-trippel $\langle \phi \rangle \text{Copy} \langle \psi \rangle$, dvs ange ett förvillkor ϕ och ett eftervillkor ψ . Specifikationen borde förklara hur programmet `Copy` kan användas utan att känna till själva koden!

2. Betrakta följande program `Copy`:

E

```
a = x;
y = 0;
while (a != 0) {
    y = y + 1;
    a = a - 1;
}
```

Gäller Hoare-trippeln $\langle \text{true} \rangle \text{Copy} \langle x = y \rangle$ enligt partiell korrekthet? Och gäller den enligt total korrekthet? Motivera dina svar.

3. Betrakta följande program `Min` som används för att beräkna minimumet $\min(x, y)$ av två heltal:

C

```
if (x > y) {
    z = y;
} else {
    z = x;
}
```

Bevisa $\vdash_{\text{par}} \langle \text{true} \rangle \text{Min} \langle z = \min(x, y) \rangle$ med Hoare-logik. Presentera beviset antingen som bevisträd eller som bevistablå. Identifiera alla resulterande bevisförpliktelser och motivera varför de gäller.

Lycka till!