Göteborgs Universitet och Chalmers Tekniska Högskola
Datavetenskap

19 december 2001
INN111/TIN120

# Tentamen i Beräkningsmodeller

Onsdagen den 19 december 2001, kl 8.45 – 12.45.
Ansvarig lärare: Bengt Nordström, tel 1033
Tillåtna hjälpmedel: Inga.
Börja varje uppgift på nytt blad. Skriv endast på en sida av papperet. Varje svar skall motiveras! Den här skriftliga tentamen utgör en del (75 %) av den totala examinationen, den andra delen (dvs. 25 %) består av de inlämningsuppgifter som har delats ut under kursens gång. För årets och förra årets elever gäller alltså att summan av poängen från inlämningsuppgifterna och den skriftliga tentan skall vara minst 100 för att få godkänt på kursen. Examensvisning kommer att äga rum fredagen den 11 januari kl 11.00 i MD3. Omtentamen sker fredagen den 18 januari. Tentamensresultat och lösningar till tentan kommer att finnas tillgängligt från kursens hemsida.

1. Reducera $\lambda$-uttrycken

   (a) $(\lambda x.(\lambda y.y\ x))\ y$

   (b) $(\lambda x.(\lambda z.z\ x))\ y$

   (20)

   Svar: Vi ser att uttrycken är $\alpha$-konvertibla, om vi systematiskt byter ut alla förekomster av variabeln $y$ mot variabeln $z$ i första uttrycket så får vi det andra uttrycket. När vi reducerar det andra uttrycket får vi $(\lambda z.z\ y)$ vilket är på normalform. Om vi försöker reducera det första uttrycket direkt ser vi att vi får en variabel-kollision, argumentet $y$ får ju inte bli bundet efter substitutionen. Så man måste göra en $\alpha$-konvertering innan substitutionen.

2. (a) Hur representerar man naturliga tal som en positionerad remsa? (5)

   (b) Hur representerar man par av naturliga tal som en postitionerad remsa? (5)

   (c) Konstruera en Turing-maskin som beräknar additionsfunktionen

   $$\mathbf{add} \in \mathbf{N} \times \mathbf{N} \to \mathbf{N}$$

   (15)

   Svar: Se läroboken sid 205

3. Det finns två olika definitioner av vad det betyder att en mängd $A$ är uppräkningsbar. En tredje skulle vara att mängdens element ryms i ett hotell med oändligt många rum, ett rum för varje naturligt tal. Att mängdens element ryms skulle betyda dels (1) att varje element har minst

ett rum, samt (2) att varje rum har högst ett element (vi vill ju inte att alla element skall kunna dela på ett rum). Visa hur dessa två krav uttrycks i de två alternativa definitionerna av uppräkningsbarhet! (20)

> Svar: Den första definitionen är att $A$ är uppräkningsbar om det finns en total surjektiv funktion $f \in \mathbf{N} \to A$. Surjektivitetskravet betyder att varje element har åtminstone ett rum (varje element är bilden av något tal), att $f$ är en funktion betyder precis att varje naturligt tal avbildas av högst ett funktionsvärde, dvs varje rum har högst ett element. Den andra definitionen är att $A$ är uppräkningsbar om det finns en total, injektiv funktion $f \in A \to \mathbf{N}$, där injektiviteten uttrycker att varje funktionsvärde (rum) avbildas av högst ett element, dvs varje rum har högst ett element och totaliteten uttrycker att varje element har minst ett rum.

4. Bevisa med hjälp av ett diagonaliseringsargument att det finns beräkningsbara funktioner i mängden $\mathbf{N} \to \mathbf{N}$ som inte finns med i $\mathbf{PRF}_1$, mängden av primitivt rekursiva funktioner med ett argument! (30)

> Svar: Vi kan räkna upp alla program i $\mathbf{PRF}_1$, låt $P \in \mathbf{N} \to \mathbf{PRF}_1$ vara en sådan uppräkning. Vi antar att $P(i)$ beräknar den matematiska funktionen $f(i) \in \mathbf{N} \to \mathbf{N}$, för alla $i$. Men nu ser vi att den beräkningsbara funktionen $\mathbf{diag} \in \mathbf{N} \to \mathbf{N}$ som är definierad av $\mathbf{diag}(i) = (f(i))(i) + 1$ inte kan vara med i uppräkningen (den skiljer sig från funktionen $f(i)$ i argumentet $i$, för varje $i$, eftersom varje funktion $f(i)$ är total (vilket beror på att varje program i $\mathbf{PRF}$ terminerar)). Att $\mathbf{diag}$ är beräkningsbar ser man eftersom man kan räkna ut dess värde i punkten $j$ genom att lägga till 1 till värdet av den beräkningsbara funktionen $f(j)$ i punkten $j$.

5. Definiera ett program $\mathbf{Y}$ i språket $\chi$ som är sådant att

$$\mathbf{Y}\ f = f(\mathbf{Y}\ f)$$

för varje funktion $f$. Bevisa att så är fallet! (30)

> Svar: Givet en funktion $f$ vill vi hitta en fixpunkt till $f$, dvs ett $x$ så att $x = f\ x$. Den typen av ekvation kan vi lösa genom att låta $x$ vara programmet $\mathbf{rec}\ x = f\ x\ \mathbf{end}$. Vi vill alltså att följande ekvation skall gälla:
>
> $$\mathbf{Y}\ f = (\mathbf{rec}\ x = f\ x\ \mathbf{end})$$
>
> dvs vi kan definiera $\mathbf{Y}$ genom
>
> $$\mathbf{Y} = \lambda f.\mathbf{rec}\ x = f\ x\ \mathbf{end}$$
>
> . Vi ser ju att uttrycket $\mathbf{Y}\ f$ beräknas till värdet av $\mathbf{rec}\ x = f\ x\ \mathbf{end}$ som beräknas till värdet av $f(f\ \mathbf{rec}\ x = f\ x\ \mathbf{end})$ vilket är lika med värdet av $f(\mathbf{Y}\ f)$.

6. Definiera ett program i $\chi$ som fungerar som operatorn för primitiv rekursion över naturliga tal, dvs konstruera ett program **natrec** för vilka följande likheter gäller:

$$
\begin{aligned}
\textbf{natrec}\ \mathsf{zero}()\ \mathrm{d}\ \mathrm{e} &= \mathrm{d} \\
\textbf{natrec}\ (\mathsf{succ}(\mathrm{x}))\ \mathrm{d}\ \mathrm{e} &= \mathrm{e}\ \mathrm{x}\ (\textbf{natrec}\ \mathrm{x}\ \mathrm{d}\ \mathrm{e})
\end{aligned}
$$

(30)

Svar: Vi kan skriva om likheterna som

$$
\begin{aligned}
\textbf{natrec}\ \mathsf{zero}() &= \lambda \mathrm{d}.\lambda \mathrm{e}.\mathrm{d} \\
\textbf{natrec}\ (\mathsf{succ}(\mathrm{x})) &= \lambda \mathrm{d}.\lambda \mathrm{e}.(\mathrm{e}\ \mathrm{x}\ (\textbf{natrec}\ \mathrm{x}\ \mathrm{d}\ \mathrm{e}))
\end{aligned}
$$

Dessa ekvationer kan lösas genom mönstermatchning följt av rekursion och vi får följande uttryck för **natrec**:

$$
\begin{aligned}
\textbf{rec}\ \mathrm{n} = \lambda \mathrm{m}.\textbf{case}\ \mathrm{m}\ \textbf{of}\ \{ \\
&\mathsf{zero}()\ : \lambda \mathrm{d}.\lambda \mathrm{e}.\mathrm{d} \\
&\mathsf{succ}(\mathrm{x}) : \lambda \mathrm{d}.\lambda \mathrm{e}.(\mathrm{e}\ \mathrm{x}\ (\mathrm{n}\ \mathrm{x}\ \mathrm{d}\ \mathrm{e}))\}\ \textbf{end}
\end{aligned}
$$

Lycka till!