INSTITUT FOR MATEMATIK OG DATALOGI

Skriftlig eksamen

D M 2 2

Lørdag den 1. juni 2002 kl. 9.00 – 13.00

Opgavesættet består af 3 sider inklusive denne forside, og indeholder 4 opgaver. Opgaverne vægtes på følgende måde:

Opgave 1:	20 %
Opgave 2:	25 %
Opgave 3:	30 %
Opgave 4:	25 %

Det bemærkes at spørgsmålene indenfor de enkelte opgaver ikke nødvendigvis vægtes lige meget.

Alle skriftlige hjælpemidler samt brug af lommeregner er tilladt. Der må gerne henvises til resultater fra lærebøgerne, men henvisninger til andre bøger er ikke acceptable som led i besvarelsen.

Opgave 1 (20 %)

I et første forsøg på at definere i Prolog et prædikat `max(L,X)` der kan bestemme det maksimale element i en liste `L` af heltal, er man kommet op med følgende forslag:

```
max([X],X).
max([X|Rest],Max):-
   max(Rest,MaxRest), (MaxRest > X ,! ,Max=Maxrest; Max=X).
```

1.a: Håndsimuler udførelsen af kaldet: "`?- max([3,2,1],X).`", idet der svares med "`;`" indtil fortolkeren svarer "`no`". Dokumenter udførelsen f.eks. ved at anvende indrykning for de indlejrede prædikatkald (call - exit, redo - exit)

Det vil dog være hensigtsmæssigt at anvende halerekursion, hvilket som bekendt kan implementeres særligt effektivt.

1.b: Definer derfor et nyt prædikat for `max(L,X)` idet der nu anvendes halerekursion. Det skal fungere på lister af vilkårlige heltal, og skal lige som den ovenstående ikke returnere en værdi ved kaldet `max([],M)`.

Opgave 2 (25 %)

2.a Reducer følgende logiske udtryk i sædvanlig prædikatnotation:

$$\exists X(p(X) \Rightarrow \forall Y(p(Y) \Rightarrow p(f(X,Y)))) \wedge (\forall Z(q(X,Z) \Rightarrow p(Z)))$$

til konjunktiv normalform og videre til clausal normalform, idet alle skridt dokumenteres.

2.b: Definer Prolog prædikatet:

```
unifiable(List1,Term,List2)
```

hvor `List2` er liste af alle de led i `List1` som matcher `Term`. For eksempel:

```
?- unifiable([X,b,t(Y)],t(a),List).
```

resulterer i:

```
List=[X,t(Y)]
```

Bemærk at `X` og `Y` her skal forblive uinstantierede, trods match med `t(a)`.
(Vink: Brug `not(Term1=Term2)`. Hvis `Term1=Term2` lykkes så fejler `not(Term1=Term2)` og den resulterende instantiering bliver annulleret)

Opgave 3 (30 %)

3.a Givet Haskell funktionen:

```
mpa f s []          = (s, [])
mpa f s (x:xs)      = (sss, y:ys)
                          where (ss, y ) = f s x
                                (sss,ys) = mpa f ss xs
```

angiv signaturen (typen) og forklar funktionens funktionalitet.

Opgave 3 og -sættet fortsættes

3.b Man kan definere multiplikation ved gentagen addition som:

```
mult n 0 = 0
mult n m = mult n (m-1) + n
```

Den "russiske bonde-multiplikation" defineres ved:

```
russ n 0 = 0
russ n m
     | (m `mod` 2 == 0) = russ (n+n) (m `div` 2)
     | otherwise        = russ (n+n) (m `div` 2) + n
```

Vurder tidskompleksiteten af begge algoritmer, idet addition, modulo 2 og division med 2 alle antages at tage tid 1. Vurder ligeledes pladskravet, og foreslå eventuelle forbedringer der kan reducere dette (det er tilstrækkeligt at angive ideerne hertil).

3.c Vis ved induktion at der for alle endelige lister `xs` gælder

```
filter p (filter q xs) = filter (p &&& q) xs
```

hvor operatoren `&&&` er defineret ved

```
p &&& q = \x -> (p x && q x)
```

Opgave 4 (25 %)

Denne opgave drejer sig om beregninger på polynomier i Haskell. Vi repræsenterer et polynomium

$$A(x) = a_0 + a_1 x + a_2 x^2 \cdots + a_n x^n$$

ved listen af dets koefficienter $[a_0, a_1, a_2, \cdots, a_n]$.

4.a Skriv en Haskell funktion `polAdd :: [a] -> [a] ->[a]` der adderer to polynomier $A(x)$ og $B(x)$, hvilket som bekendt gøres ved at addere koefficienterne. Husk at de to operander ikke nødvendigvis har samme grad (det samme antal koefficienter).

4.b Skriv en Haskell funktion `polEval:: [a] -> a -> a` der for givet x beregner værdien af et polynomium $A(x)$ i lineær tid ($\mathcal{O}(n)$). Vink: Værdien af et polynomium kan beregnes ved *Horners skema*:

$$A(x) = (\ldots((a_n)x + a_{n-1})x + \cdots)x + a_0.$$

4.c Skriv en Haskell funktion `polMul :: [a] -> [a] ->[a]` der multiplicerer to polynomier $A(x) = a_0 + a_1 x + a_2 x^2 \cdots + a_n x^n$ og $B(x) = b_0 + b_1 x + b_2 x^2 \cdots + b_n x^m$, idet produktet $C(x) = c_0 + c_1 x + c_2 x^2 \cdots + c_{n+m} x^{n+m}$ har koefficienterne:

$$c_k = \sum_{i+j=k} a_i b_j$$

for $k = 0, 1, \cdots, n + m$.

Opgavesættet slutter her