

DIKUrevy 2000

Kvindelogikprogrammering

skrevet af Uffe og folk

Status: næsten færdig – skal skrives til 2 personer

(7 min.)

Roller:

F1 (Uffe)	Forelæser
F2 (Christoffer)	Forelæser

-
- F** : Vi har I mange år forsøgt at tiltrække flere kvinder her på datalogi. Det har vi gjort fordi... well... (*fjerner "savl" fra hagen*) fordi det synes vi kunne være fedt.
- F** : Nu ligger DIKUs udspil så endelig klar. Ved at tage udgangspunkt i logikprogrammering har vi opfundet begrebet "kvindelogikprogrammering" – også kendt som ulogisk programmering.
- F** : Vi har konkretiseret dette i en pendant til sproget Prolog. Det har vi valgt at kalde "Kontrallog".
- F** : Kontrallog er dog en smule anderledes end de fleste programmeringssprog vi kender. Tag for eksempel tildelingssætningen "a lig med 3". Denne sætning vil i nogle få tilfælde have effekten at a tildeles værdien 3. I Kontrallog vil a dog i de fleste tilfælde blive tildelt en vilkårlig anden værdi. Enkelte gange kan a sågar helt forsvinde.
- F** : Ligeledes findes naturligt nok udviklinger af de velkendte betingede sætninger. Vores "if then else" konstruktion får en tredelt udgave i konstruktionen "if then else otherwise". Dette kan dog udskrives til en "if perhaps", der kan følges vilkårligt senere i programmet af en matchende "unless otherwise".
- F** : Det skal nævnes at der naturligt nok ikke findes variable i Kontrallog, men kun konstanter. Sprogets indeterministiske natur gør dog dette til en mindre hindring end man ellers skulle tro. Alt er naturligt nok dynamisk og man finder derfor ingen statiske erklæringer – enhver værdi er af sin natur volatile.
- F** : De primitive typer i Kontrallog er blevet forenklet i forhold til gængse programmeringssprog. Alle de "besværlige typer" er afskaffet, såsom heltal og tegn. Tilbage er kun flydende kommatal til at opbevare de mere reelle værdier og så selvfølgelig strenge, ja til tider "helt vildt strenge", som til gengæld er udvidet til at kunne klare vilkårligt lange samtaler.
- F** : Kontrallog indeholder selvfølgelig også en videreudvikling af tidens trend: objektorienteret programmering. Objekterne bliver til subjekter og de har hverken metoder eller egenskaber, men fornemmelser og følelser. Disse pakkes naturligt nok ind, i visse tilfælde, og man skal således ikke regne med at få lov til at give argumenter til et subjekts private følelser.
- F** : For at føre parallellen fra objekterne helt over i subjekterne har et subjekt brug for hvad der svarer til konstruktør og destruktør. Her har vi indført en kontrollør og en inspektør – de såkaldte "uniforme". Hvis både kontrollør og inspektør mangler er der mulighed for såkaldte suspekte subjekter.
- F** : Alt dette tænkes kørt på en virtuel maskine – hvad ellers? – hvor *alt* fortolkes. Ofte bliver det endda kun partielt evalueret. Denne maskines byggesten – dvs. bytekoden eller maskinkoden om man vil – har tre basisfaktorer: 1) "fordi jeg siger det", 2) "derfor" og 3) catch-all-casen "sådan er det bare".

- F** : Denne virtuelle maskine er desuden udstyret med en uendelig stor cache. Ved enhver beslutningssætning sammenlignes den igangværende evaluering med tidligere evalueringer, og man kan ofte ende med resultatet “bare ærgerligt” eller “der kan du bare se!” Enhver reference til denne cache fra programmørens side har i praksis vist sig at få fortolkeren til at gå ned. Cachen kan ikke flushes eller på anden måde slettes – alt bliver husket!
- F** : Det eneste begreb som har overlevet transformationen fuldstændigt intakt er “kommando promptet”. *Det* var faktisk noget kvinderne var meget glade for.
- F** : Enkelte datastrukturer er også overlevet næsten intakt. F.eks. har træet ikke ændret sig meget, selvom kvinderne måske synes det er noget rod og prøver at kante sig uden om. De slår knuder. Der findes stadig tre forskellige slags gennemløb, de har bare fået andre navne: præ(menstruel)order, out of order og sidste, men ikke mindst, postordre. . . også kendt som TV-shop.
- F** : Listerne overlever også næsten intakt, selvom vi endnu ikke har afdækket fuldt hvad konsekvensen bliver af at erstatte alle sorteringsalgoritmerne med randomiseringsalgoritmer.
- F** : Et af de endnu uafdækkede problemområder i kvindelogikprogrammering generelt er at vi oplever periodemæssigt svingende køretidskompleksitet, hvor der oftest er udelelig adgang til hele objektet, og udsultning kan forekomme. Vi ser dog frem til at se hvilke fordele kollektiv garbage collection på toilettet kan give os, og glæder os til at kaste os over problemet med “De Anorætiske Filosofer”.
- F** : Til slut skal nævnes at vi naturligvis i hele processen har været stærkt inspirerede af vores matematiske kolleger, der har forsøgt at definere kvinden som “ $R \text{ p\aa} Q$ ” . . . altså alt det irrationelle. De arbejder netop nu på en hypotese om “ $C \text{ p\aa} Q$ ” . . . altså alt det komplekse og irrationelle. Mon ikke de også når frem til at det er “ $C \text{ p\aa} R$ ”? Altså alt det komplekse og imaginære der ikke er reelt.
- F** : Vores næste projekt er så pendanten til Miranda: ufunktionel programmering i Amanda. Tak for i aften!

Tæppe