

# SI-möte #1, Programkonstruktion

Elias Castegren (*Några uppgifter lånade från Carl Leonardsson*)

elca7381@student.uu.se

3 november 2010

## Begrepp

Vad betyder följande begrepp? Beskriv och ge exempel! (Du behöver inte ge formella definitioner)

*i)* Uttryck    *ii)* Värde    *iii)* Typ    *iv)* Funktion    *v)* Invariant    *vi)* Tupel

## Övningar

### 1.

Vad har nedanstående SML-uttryck för värden? Vilka typer har värdena? Om något inte går att beräkna, berätta varför.

*i)* `2 * 3+4`    *ii)* `true orelse false`    *iii)* `false orelse true`

*iv)* `size("Hej " ^ "hopp!")`    *v)* `if true then 42`

*vi)* `if 2>3 then 2.0 else (3.14, true)`

*vii)* `(4 < 7, (12.0 / 3.0, 101 mod 10))`

*viii)* `13 mod 4 + round(if 4>2 andalso false then 3.3 else 3.5)`

### 2.

Låt en rätvinklig låda representeras av längden av sina sidor  $i$ ,  $j$  och  $k$  (som alla är av typen `int`).

*i)* Skriv ett *uttryck* som beräknar lådans volym.

*ii)* Skriv en *funktion* som, given sidlängderna som argument, beräknar lådans volym.

*iii)* Finns det någon lämplig invariant att påtvinga lådrepresentationen  $i$ ,  $j$ ,  $k$ ?

### 3.

Det formella sättet att skriva att ett värde  $x$  har typen  $t$  är  $x \in t$ . Vanligtvis skriver man dock istället  $x : t$ . Detta är även SML-syntax för att tvinga t.ex. ett funktionsargument att ha en viss typ. Skriv funktioner som ger:

- i) Volymen av en kub med sidan  $x : \text{int}$ .
- ii) Sant om strängen  $s : \text{string}$  består av minst 3 tecken, falskt annars.
- iii) Det största talet av  $x : \text{real}$  och  $y : \text{real}$  (Hur kan man göra om  $x : \text{int}$  och  $y : \text{real}$ ?)
- iv) Sant om  $n : \text{int}$  är ett jämnt tal större än 42, falskt annars.
- v) Absolutbeloppet,  $|z|$  av  $z : \text{real}$ .
- vi) `true` om *exakt en* av  $a : \text{bool}$  och  $b : \text{bool}$  är `true`, `false` annars.

Vad har funktionerna för typer? Fundera på när du behöver berätta för ML vilken typ argumenten ska ha och när MLs typhärledning automatiskt ger funktionen den efterfrågade typen.

### 4.

Fyll i resultatet av varje anrop av funktionen `foo`.

```
val a = 5;
fun foo(x) = x div a;
foo(17);    →
val a = 0;
foo(13);    →
fun foo(x) = x mod a;
foo(13);    →
```

### 5.

Betrakta nedanstående funktion:

```
fun nanook(alphonzo, fatherOblivion) =
  if size alphonzo < size fatherOblivion then
    alphonzo ^ String.substring(fatherOblivion,
      size alphonzo, size fatherOblivion - size alphonzo)
  else
    alphonzo
```

Vad har den för typ? Vad beräknar funktionen?

Hur beräknas anropet `nanook("Struts", "Fläsklägg")` (visa beräkningen steg för steg)?

---

*Lycka till!*