

## 1 Begrepp

Vad betyder begreppen? Beskriv och ge exempel!

- a) Abstrakt datatyp
- b) Invariant
- c) Primitiver

## 2 Övningar

- a) Vad är det för skillnad på en datatyp och en abstrakt datatyp? Fördelar/nackdelar?

Antag att du har definierat en abstrakt datatyp för hantering av bilar:

```
abstype car = Car of string*real*bool*real (* Namn, Maxhastighet, Turbo?, Pris i kr *)
with
  (* Konstruktörer *)
  val NewCar = Car("No Name",0.0,false,0.0)
  fun setName(newname,Car(_, b, c, d)) = Car(newname, b, c, d)
  fun setSpeed(newspeed,Car(a, _, c, d)) = Car(a, abs(newspeed), c, d)
  fun enableTurbo(Car(a, b, _, d)) = Car(a, b, true, d)
  fun setPrice(newprice,Car(a, b, c, _)) = Car(a, b, c, newprice)

  (* Selektörer *)
  fun getName(Car(name, _, _, _)) = name
  fun getSpeed(Car(_, speed, _, _)) = speed
  fun hasTurbo(Car(_, _, turbo, _)) = turbo
  fun getPrice(Car(_, _, _, price)) = price
end
```

Skriv SML-funktioner enligt specifikationerna. **Observera!** Funktionerna kommer ligga utanför den abstrakta datatypen:

- b) `car2tuple(c) : car -> string*real*bool*real`. Returnerar innehållet hos `c` som tuple.
- c) `filterTurbo(x) : car list -> car list`. Returnerar `x` fast alla bilar utan turbo är borttagna.
- d) `createCar(n, s, t, p) : string*real*bool*real -> car`. Skapar en `car`-värde där namnet blir `n`, hastigheten blir `s`, turbo blir `t` och priset blir `p`.