

Übungen zu Grundlagen der Softwarezuverlässigkeit

Abgabe bis zum 18.02.05

Hinweis:

Die Übung am 11.02. wird auf den 18.02. nach hinten verlegt, dafür wird die Vorlesung vom 18.02. bereits am 11.02. gehalten werden. Die Uhrzeiten bleiben jedoch unverändert (V: 11:30-13:00, Ü: 14:00-15:30). Wie gewünscht wird die Übung in V38.02 stattfinden.

Aufgabe 7.1

Es sei

$$C = \{0 : 00, 0 : 01, \dots, 0 : 59, 1 : 00, \dots, 23 : 59\}$$

die Menge von "Uhrzeiten" und die folgende Abstraktion

$$A_0 = \{\text{Morgen, Vormittag, Mittag, Nachmittag, Abend, Nacht, } \perp, \top\}$$

mit den Konkretisierungen

$$\begin{aligned} \gamma(\text{Morgen}) &= \{7 : 30, \dots, 10 : 30\} \\ \gamma(\text{Vormittag}) &= \{10 : 00, \dots, 12 : 00\} \\ \gamma(\text{Mittag}) &= \{11 : 30, \dots, 14 : 00\} \\ \gamma(\text{Nachmittag}) &= \{15 : 00, \dots, 16 : 45\} \\ \gamma(\text{Abend}) &= \{17 : 30, \dots, 22 : 00\} \\ \gamma(\text{Nacht}) &= \{23 : 00, \dots, 23 : 59\} \cup \{00 : 00, \dots, 6 : 00\} \\ \gamma(\perp) &= \emptyset \\ \gamma(\top) &= C \end{aligned}$$

gegeben.

- Erweitern Sie A_0 durch möglichst wenig zusätzliche Elemente zu einem vollständigen Verband A .
- Geben Sie ein Beispiel dafür an, dass die Operatoren \sqcap und \sqcup auf (nicht wohlgeformten) A und auf 2^C i.A. nicht übereinstimmen, d.h. geben Sie eine Teilmenge $S \subseteq A$ an mit $\gamma(\sqcap S) \neq \sqcap \{\gamma(a) \mid a \in S\}$ (entsprechend für \sqcup).
- Falls A nicht wohlgeformt ist, so ändern Sie A entsprechend ab.
- A wird um die Elemente Uni, a.m., p.m. mit den Konkretisierungen

$$\begin{aligned} \gamma(\text{Uni}) &= \{8 : 00, \dots, 17 : 15\} \\ \gamma(\text{a.m.}) &= \{0 : 00, \dots, 11 : 59\} \\ \gamma(\text{p.m.}) &= \{12 : 00, \dots, 23 : 59\} \end{aligned}$$

zu A' erweitert.

Überprüfen Sie, ob A' noch immer ein wohlgeformter, vollständiger Verband ist und passen Sie A' notfalls an.

- (e) Zusatzaufgabe: Warum stimmt für wohlgeformtes A jedoch \sqcap mit \cap im Sinn von Teilaufgabe b überein? Das heißt, zeigen Sie, für jede Menge $S \subseteq A$ gilt

$$\gamma(\sqcap S) = \cap\{\gamma(a) \mid a \in S\}.$$

Aufgabe 7.2

Betrachten Sie folgenden Ausschnitt aus einem Programm:

```
var
  angle    : float ;
  pi       : float ;
  deg2arc  : float ;
  res      : float ;
begin
  pi       := 3.1415926535898;      (l0)
  deg2arc  := 2 * pi / 360;        (l1)
  res      := sin( angle * deg2arc ); (l2)
end;                               (l3)
```

Es ist wünschenswert, dass der Compiler automatisch konstante Werte erkennt, diese in dem erzeugten Code durch entsprechende Konstanten ersetzt und somit überflüssige Berechnungen und Variablen einspart.

Hierfür soll der Wert einer Variable durch einen Wert aus

$$A_0 = \{(\text{konstant}, v) \mid v \in \mathbb{R}\} \cup \{\text{dynamisch}\}$$

abstrahiert werden. Die Konkretisierung ist durch $\gamma_0((\text{konstant}, v)) = \{v\}$ und $\gamma_0(\text{dynamisch}) = \mathbb{R}$ gegeben.

- Ist A_0 bereits ein wohlgeformter, vollständiger Verband? Erweitern Sie A_0 notfalls.
- Wie muss die entsprechende Abstraktionsabbildung α_0 für A_0 gewählt werden?
- Die Variablen seien entsprechend der Vorlesung (vgl. Folie 341) zu einem Vektor $(\text{angle}, \text{pi}, \text{deg2arc}, \text{res})$ aus $C = \text{float}^4$ zusammengefasst, welcher durch Werte aus $A = A_0^4$ komponentenweise abstrahiert wird. Entsprechend ergeben sich γ , α und \sqsubseteq komponentenweise aus γ_0 , α_0 und \sqsubseteq_0 .

Geben Sie entsprechend der Vorlesung die abstrakten Zuweisungen und Filter für die obigen Anweisungen an.

- Berechnen Sie nun die abstrakten Werte aller Variablen bei l_0, \dots, l_3 entsprechend der Vorlesung. Mit welchem abstrakten Wert sind alle Variablen zu Beginn initialisiert?

Aufgabe 7.3

- Unter Verwendung der Abstraktion aus der letzten Aufgabe berechnen Sie die abstrakten Werte der Variablen an jeder Stelle des folgenden Programms mit Hilfe der Sammelsemantik (vgl. Folie 357ff).

```

var
  x      : integer;
  y      : integer;
  z      : integer;
begin
  z := 13;          (l0)
  if x = 11 then   (l1)
    y := 2;        (l2)
  else
    z := 14;       (l3)
  end;
  if y > 4 then    (l4)
    y := 2 * z;    (l5)
  else
    x := 3 * y;    (l6)
  end;
end;                (l7)

```

- (b) Gehen Sie wie in der letzten Teilaufgabe vor, berechnen Sie jedoch zum Vergleich die Werte auch ohne die Sammelsemantik (vgl. Folie 345).

```

var
  x      : integer;
  y      : integer;
begin
  if isEven( x ) then          (l0)
    y := 2;                    (l1)
  else                          (l2)
    y := -2;                   (l3)
  end;
  x := x * ( y - 2 ) * ( y + 2 ); (l4)
  if x = 0 then                (l5)
    y := 1;                    (l6)
  else                          (l7)
    y := 3;                    (l7)
  end;
end;                            (l8)

```

Wie könnte man die Abstraktion abändern, so dass die Resultate beider Verfahren für dieses Programm übereinstimmen?