

Übungen zur Vorlesung  
Bäume, Ordnungen und Anwendungen  
Blatt 5

Prof. Dr. Roland Meyer

Abgabe bis 01.12.2015 um 14h

**Aufgabe 5.1** (Funktionenverbände)

Sei  $(D, \leq)$  ein vollständiger Verband und sei  $F_D := \{f : D \rightarrow D \mid f \text{ ist monoton}\}$  die Menge aller monotonen Funktionen über  $D$ . Es gelte  $f_1 \preceq f_2$  genau dann, wenn  $f_1(d) \leq f_2(d)$  für alle  $d \in D$ .

Zeigen Sie, dass  $(F_D, \preceq)$  ein vollständiger Verband ist.

**Aufgabe 5.2** (Procedure Summaries)

Betrachten Sie folgendes Programm.

<pre> <b>proc</b> [main()]<sup>1</sup>   [g := 2]<sup>2</sup>   [l := g]<sup>3</sup>   [work()]<sup>4</sup>   [ret := 1 - ret]<sup>6</sup> <b>end</b> </pre>	<pre> <b>proc</b> [work()]<sup>7</sup>   [l := g]<sup>8</sup>   <b>if</b> [l &gt; 0]<sup>9</sup> <b>then</b>     [work()]<sup>10</sup>   [ret := l]<sup>12</sup> <b>end</b> </pre>
--	--

Führen Sie eine *Copy-Propagation-Analyse* durch. Gesucht ist für jeden Block die Menge der Variablen, die am Ausgang definitiv den gleichen Wert wie  $g$  haben (vorwärts-must-Analyse). Zum Beispiel hat  $ret$  nach Block 12 den gleichen Wert wie  $g$ , nicht aber nach Block 6.

Benutzen Sie den Verband  $(\mathbb{P}(Var), \supseteq)$ , mit  $Var = \{g, ret, l\}$ . Benutzen Sie ferner die Menge der globalen Variablen  $Glob = \{g, ret\}$ .

Gehen Sie wie folgt vor.

- a) Stellen Sie die Transferfunktionen auf, welche Sie für die Berechnung einer Procedure Summary für die *work()*-Prozedur benötigen. Geben Sie insbesondere  $f_{call}$  und  $f_{return}$  an.

*Hinweis: Sie benötigen  $f_{l0}$  und  $f_{l1}$  nicht, da diese durch  $f_{call}$  und  $f_{return}$  abgedeckt werden.*

- b) Stellen Sie das Summary-Gleichungssystem auf.

*Hinweis: Sie benötigen kein  $X_{l1}$ , wenn Sie strikt der Definition von  $callret(\cdot)$  aus der Vorlesung folgen.*

- c) Vereinfachen Sie ihr Gleichungssystem so weit wie möglich, um sich in der folgenden Teilaufgabe etwas Arbeit zu sparen.

- d) Lösen Sie das Summary-Gleichungssystem durch Fixpunkt-Iteration auf dem Verband der monotonen Funktionen.

*Hinweis: Stellen Sie eine Funktion durch ihre Funktionsvorschrift  $Y_i(X)$  dar.*

- e) Schließen Sie die Datenflussanalyse, unter Verwendung der soeben erstellten Procedure-Summary, ab.

**Aufgabe 5.3** (Procedure Summaries und *call-return*-Verhalten)

Im Programm aus Aufgabe 5.2 wird die Variable  $g$  nach ihrer ersten Belegung nicht mehr verändert. Dadurch kann man sicher sein, dass alle Kopien von  $g$  auch nach der Ausführung von *work()* noch Kopien von  $g$  sind. Betrachten Sie nun das folgende Programm mit einer modifizierten *work()*-Prozedur:

```

proc [main()]1
  [g := 2]2
  [l := g]3
  [work()]4
  [ret := 1 - ret]6
end

proc [work()]7
  if [g > 0]9 then
    [work()]1011
  else
    [g := g - 1]12
  [ret := g]13
end

```

Hier wird in der *work()*-Prozedur die Variable  $g$  geändert. Damit ist am Ende der Ausführung auch  $l$  (in *main()*) keine Kopie von  $g$ . Adaptieren Sie die Analyse so, dass diese Situation erkannt wird und das richtige Ergebnis geliefert wird.

*Hinweis: Um zu erkennen, ob  $g$  in der Unterfunktion geändert wurde, kann *call* eine Hilfsvariable einführen, die beim Rückspurng durch *return* ausgewertet wird.*

**Abgabe bis 01.12.2015 um 14h im Kasten neben Raum 34-401.4**