Lecture: Program analysis
Exercise 8
http://proglang.informatik.uni-freiburg.de/teaching/programanalysis/2010ss/

Abstract interpretation

1 Widening operators

Show that the operator ∇ on **Interval** with

$$\bot \nabla X = X \nabla \bot = X$$

and

$$[i_1, j_1] \nabla [i_2, j_2] = [\text{ if } i_2 < i_1 \text{ then } -\infty \text{ else } i_1,\ \text{if } j_2 > j_1 \text{ then } +\infty \text{ else } j_1]$$

is a widening operator. First, state precisely what you need to show, and then show that these properties are indeed fulfilled.

2 Abstractions

Let S be the set of strings over a (finite) alphabet Σ. An abstraction of the string is the set of characters/symbols of which the string is built. Example: `Program analysis` is abstracted by {`P,r,o,g,a,m, ' ',n,l,y,s,i`}.

Specify the details of the Galois connection $(\mathcal{P}(S), \alpha, \gamma, \mathcal{P}(\Sigma))$ formally. Is this Galois connection also a Galois insertion?

3 Galois insertions

Let $(L_1, \alpha_1, \gamma_1, M_1)$ and $(L_2, \alpha_2, \gamma_2, M_2)$ be Galois insertions. First define

$$\begin{aligned} \alpha(l_1, l_2) &= (\alpha_1(l_1), \alpha_2(l_2)) \\ \gamma(m_1, m_2) &= (\gamma_1(m_1), \gamma_2(m_2)) \end{aligned}$$

and show that $(L_1 \times L_2, \alpha, \gamma, M_1 \times M_2)$ is a Galois insertion. Then define

$$\begin{aligned} \alpha(f) &= \alpha_2 \circ f \gamma_1 \\ \gamma(g) &= \gamma_2 \circ g \alpha_1 \end{aligned}$$

and show that $(L_1 \to L_2, \alpha, \gamma, M_1 \to M_2)$ is a Galois insertion.

4 Types and Effects

Consider the following FUN program:

```
new_A x := 1 in
new_B y := 9 in
let f = fn z => x := !y in
let g = fn z => x := 8 in
let h = fn z => !x in
(fn w => w f + w h) (fn v => v 4)
```

What is the result of evaluating this program? What are the types and effects for the functions in this program?

5 Control Flow Analysis in a Type and Effect System

The type and effect system for Control Flow Analysis in Chapter 5.1. uses annotations ϕ to denote the set of function definitions that can result in a function of a given type.

Extend the analysis with annotations for the base type `bool` to denote the set of constants that may be the result of evaluating the expression of a respective type.

Submission

- Deadline: 20.07.2010, 11:00, per mail to `bieniusa@informatik.uni-freiburg.de`, or on paper to Annette Bieniusa, Geb. 079, Room 000-14.
- Late submissions will not be marked.
- Do not forget to put your name on the exercise sheet.