Design and Analysis of Real-Time Systems SS 2013

Jan Reineke
Andreas Abel

Deadline: Thursday, May 23, 2013, 14:15

Assignment 4

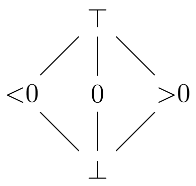
On this assignment sheet, we will use the *PAG Program Analyzer Generator* (`http://pag.cs.uni-sb.de/`) to implement different program analyses. Please submit your solutions for the practical parts of this assignment to `abel@cdl.uni-saarland.de`.

Problem 1: Preparation (0 Points)

Make yourself familiar with *PAG*, e.g., by reading the "How to build your own analysis" section on the above mentioned website, or by playing around with the *parity analysis* which was shown in the lecture and is available on the course website.

Problem 2: Simple Sign Analysis (3+2+10 Points)

In this exercise, you will implement a simple sign analysis using the following lattice. (Note that this analysis is different from the sign analysis on the previous assignment sheet.)



1. Derive the abstract operators for addition, subtraction, multiplication and division:

$+^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

$-^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

$*^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

$/^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

2. Derive the abstract operators for $<$ and $=$. Explain how the abstract operators for $\leq$, $\geq$, $>$, and $<>$ can be obtained from these operators.

$<^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

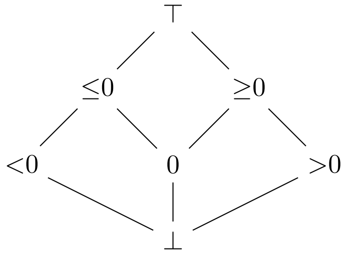
$=^\#$	⊥	<0	0	>0	⊤
⊥					
<0					
0					
>0					
⊤					

3. Implement this analysis! You can do this by modifying the *parity analysis* accordingly. You might find it helpful to use a flat lattice of integers as your abstract domain, and to represent <0 by -1, 0 by 0, and >0 by 1.

   Run the analysis on the program from Problem 3.5 of the previous assignment and report the results (you can submit a screenshot showing the results).

Problem 3: Extended Sign Analysis (3+2+10 Points)

Now, we consider again the sign analysis from the previous assignment sheet:



1. Derive the abstract operators for multiplication and division:

$*^{\#}$	⊥	0	<0	≤0	>0	≥0	⊤
⊥							
0							
<0							
≤0							
>0							
≥0							
⊤							

$/^{\#}$	⊥	0	<0	≤0	>0	≥0	⊤
⊥							
0							
<0							
≤0							
>0							
≥0							
⊤							

2. Derive the abstract operators for < and =. Explain how the abstract operators for ≤, ≥, >, and <> can be obtained from these operators.

$<^{\#}$	⊥	0	<0	≤0	>0	≥0	⊤
⊥							
0							
<0							
≤0							
>0							
≥0							
⊤							

$=^{\#}$	⊥	0	<0	≤0	>0	≥0	⊤
⊥							
0							
<0							
≤0							
>0							
≥0							
⊤							

3. Implement this analysis!

   Hints: You might use a lattice over the power set of the integers. In that case, you will also have to modify the *combine function.* To this end, you might want to take a look at *ZF expressions* in the PAG documentation. You might also find the *member test* operator "?" helpful.

   Run the analysis on the program from Problem 3.5 of the previous assignment and report the results (you can submit a screenshot showing the results).

Note: PAG expects *if-then-else* expressions to end with *endif*; otherwise, it will complain about an "unexpected token".