Design and Analysis of Real-Time Systems SS 2013

Jan Reineke
Andreas Abel

Deadline: Thursday, June 20, 2013, 14:15

Assignment 7

Problem 1: Must Analysis (8·4 Points)

On this assignment, we will formalize the *must cache analysis* for LRU caches that was introduced informally in the previous lectures. The goal of this assignment is to help you understand "the big picture" of the material discussed in class so far. Therefore, we deliberately provide only little guidance on how to attack this problem. This also means that many correct solutions are possible, and grading will be lenient.

In the following, we assume that M denotes the set of memory blocks, and ϵ an empty cache line.

1. Find a domain C that can represent concrete cache states of a fully-associative LRU cache with associativity k. Note that a concrete cache state of an LRU cache needs to have information about which blocks $m \in M$ are in the cache as well as the recency of their last access.
2. Find a domain A that can represent abstract cache states for the *must analysis*.
3. Define the concretization function $\gamma : A \to \mathcal{P}(C)$.
4. Define the abstraction function $\alpha : \mathcal{P}(C) \to A$.
5. Define an order $(\sqsubseteq_A) \subseteq (A \times A)$ such that $(\mathcal{P}(C), \subseteq)) \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} (A, \sqsubseteq_A)$ is a Galois connection.
6. Define the abstract least-upper-bound operator $\sqcup : (A \times A) \to A$ explicitly, i.e., without refering to the order $\sqsubseteq_A$.
7. Define a concrete transformer $access : C \times M \to C$ that describes the effect of a memory access on a cache state.
8. Define the corresponding best abstract transformer $access_{abs} : A \times M \to A$ explicitly, i.e., without refering to α and γ.