

Verification of Real-Time Systems SS 2015

Assignment 8

Deadline: June 25, 2015, before the lecture

Exercise 8.1: Cache Replacement Policies (1.5+1.5+1.5+1.5+2=8 Points)

In the lecture, we introduced some cache replacement policies such as LRU, FIFO, MRU, and PLRU. There are many more, e.g. *Pseudo Round-Robin* that works as follows: Upon an access to a cache set, the element in the set is replaced that resided in the *cache way* that missed least recently.

To implement a replacement policy for a k -way set-associative cache with n cache sets, a certain amount of metadata is required to express the state of the replacement policy. Derive for each of the above mentioned policies, how many bits of metadata RAM are required at least. Explain why this amount of bits is sufficient and necessary, and describe the meaning of these bits. What is special about Pseudo Round-Robin compared to LRU, FIFO, MRU, or PLRU?

Hint: A k -way associative cache comprises k so-called cache ways. The i -th cache way holds all i -th element of each cache set.

Exercise 8.2: FIFO Analysis (1+1+4+1+2+1=10 Points)

- Explain why it is useful for predicting hits in a FIFO cache to first predict misses. How does this insight manifest in FIFO cache analyses?
- Formalize the concretization function for the *Hits-after-Misses* must domain. Provide the signature of the concretization function.
- Formulate and prove local consistency of the abstract transformer of the *Hits-after-Misses* must domain in the case that the classification of the access to update is exactly known.
- Another idea for predicting cache hits in FIFO caches is based on *phases*. Briefly explain what a phase is and why phases are useful to predict hits.
- Consider the following access sequence prefix $\langle abc \rangle$. How many additional phases of the respective following type need to be detected before predicting hits for the blocks in the phases $\{c\}$, $\{bc\}$, $\{abc\}$, $\{abcd\}$. Consider the following continuations of the sequence: $\langle abcabcabcabc \rangle$ and $\langle bcbbcbbc \rangle$. Which phases need to be tracked by an analysis for each continuation?
- Consider a loop accessing the same $l \leq k$ pairwise different memory blocks in each iteration, starting with an initially unknown fully-associative FIFO cache with associativity k . Does virtual loop peeling help to predict hits for these accesses? Explain shortly.

Exercise 8.3: Persistence Analysis (1+2+3+3+2=11 Points)

We want to analyse the cache behaviour of the program below. First construct the access graph to ease the following tasks. Assume an initially empty 2-way fully-associative LRU cache.

- Perform a must/may analysis and give the results. What could you predict?
- Perform a context-sensitive must/may analysis by virtual loop peeling and give the results. What could you predict?

- (c) In the lecture, we introduced persistence analyses as a further analysis option. First perform the set-wise conflict counting analysis and give the results. What could you predict? Second perform the block-wise conflict counting analysis and give the results. What could you predict?
- (d) Could you achieve the same results as in (c), without persistence analysis but with context-sensitive must/may analysis using specific contexts? Briefly explain why this is not possible, or sketch a solution by explaining the contexts you introduce and their usage within the analysis.
- (e) Can any block be classified persistent under FIFO replacement?

```
while(...) {  
  read a;  
  if (...) {  
    read a;  
  } else if (...) {  
    read b;  
  } else {  
    read b;  
    read c;  
    read b;  
    read a;  
  }  
}
```