UG Exam Performance Feedback
Second Year
2018/2019 Semester 2

COMP25212 System Architecture

Javier Navaridas-Palma
Antoniu Pop

Comments

Q1 and Q2
Students did very well in most cases on questions that have a substantial component of textbook knowledge (Q1a and Q2a) as well as on the two questions mixing textbook knowledge with practical considerations (Q1b and Q2c). The more problem-solving focused Q1c showed that some of the finer points regarding cache behaviour, including the practical advantage of exploiting spatial locality with larger cache lines, or the importance of cache associativity to reduce conflict misses when memory accesses are strided, were less well understood. Many students focused solely on the size of the working set in comparison to that of the cache, deciding that cache misses were therefore predominantly related to capacity and failed to consider the conflict misses.

The least well-tackled question was Q2b, and more specifically sub-question (ii), where many students did not attempt to quantify the change in mean time to failure within the RAID 5 system. Furthermore, most students limited their analysis to the first disk failure whereas the second one is also necessary for this system to fail. Part (i) was generally well handled, aside from two common mistakes: calculating storage efficiency rather than storage capacity, and forgetting that not all disk capacity is used for data storage in a RAID 5 as there is also parity. In part (iii), many did not identify the key weakness of RAID 5 systems, but the majority were able to provide better alternatives nonetheless.

Q3a has generally had a good performance from students. Common mistakes were to assert that WAR and WAW dependencies arise from pipelining, when they actually come from OoO architectures. Similarly happens with structural hazards, which come from modern multi-FU architectures, not from pipelining. These were not penalised. Q3b had rather bad performance, much worse than I was expecting, specially since it was a bookwork question. The most common mistake was to discuss the differences between OoO architectures or the benefits of OoO over in-order architectures; neither of which was asked for. Among the students answering properly, only a few forgot about result management to deal with OoO dependencies.

Q4a-d had good performance, but I was expecting somewhat better results.
The main issue here has been to forget about orders of magnitude (10^n) and also the units. Some students had some mistakes in their operations or used incorrect numbers. When the working out allowed to see this was the case, I haven't penalised them.

Finally Q4e has had acceptable performance as well. Some common issues were related to the copy back nature of MESI. Students either assumed that main memory was updated with each write, or forgot to update in main memory when changing from M to S. Most concerning is the fact that some students believed that the bus is used to send messages to query other caches states before doing local changes. Message names were not always written correctly, but I haven't penalised this as the important thing is that they understand the overall idea instead of remembering the specifics.