UG Exam Performance Feedback
Second Year
2019/2020 Semester 1

COMP26120 Algorithms and Imperative Programming

Giles Reger
Lucas Cordeiro

Comments

Section A Feedback

Overall, the performance was excellent.

The performance of the asymptotic analysis was reasonable. Most candidates were able to analyse the asymptotic behaviour of the Big-O notation statements and correctly classify them. Only a few candidates were unable to analyse the mathematical properties of the given expressions, in particular, the transitivity rule.

Generally, candidates got the comparison of running times right. They were able to determine the smallest value of n such that algorithm A runs faster than algorithm B on the same machine. A minority of the candidates tried to guess the smallest value of n.

Overall, the performance to compute the code time complexity was excellent. Most candidates were able to precisely find the computational complexity of the given code fragment, which can be computed by analysing the bounds of the nested loops. Only one candidate has incorrectly guessed "none of the above" option.

The complexity analysis part of the exam was well done. Most candidates were able to analyse the performance of the best algorithm to process the specified number of items. Only a few candidates were unable to determine the performance of the insertion and merge sort algorithms.

The sum question to represent the loops was well done. Most candidates were able to correctly compute the complexity of the first and second loops using integrals. Just one candidate has incorrectly applied integrals, thus incorrectly concluding the complexity of the second loop.

Generally, candidates got the equations of the master method right. In particular, they were able to define the asymptotic bounds of T(n) correctly based on the master theorem. Only a few candidates have incorrectly guessed "none of the above" option.

Overall, the performance to use the master theorem was excellent. Most candidates were able to identify the cases of the master method to give tight asymptotic bounds for T(n). Just one candidate has incorrectly guessed "none of the above" as the master theorem case.

Overall, the performance for computing the complexity of the recurrence was excellent. Most candidates were able to apply the master method to give tight asymptotic bounds of the given recurrence equation. Just one candidate has incorrectly considered f(n) as the tight asymptotic bounds.

Section B Feedback

Overall, performance was very good.

Almost all students correctly identified the heap-order property and a priority queue from their description, and correctly identified properties of given trees.

Many students noticed that the height of a tree and the complexity of searching in a tree are unconnected. Fewer students (about two thirds) recalled the reasoning behind the linear space property of skip lists and a similar number spotted that double hashing for open addressing is unrelated to the memory efficiency of separate chaining.

Around 75% of students correctly identified whether given AVL trees were constructed correctly

The most challenging section of this part was a scenario where students needed to identify which data structures were most appropriate given different options. The correct answers depended on the students noticing that insertion into a linked list is much cheaper than insertion into an AVL tree, and that a hash table using separate chaining with a small capacity quickly degenerates into a linked list. More than half of students identified the correct answer in all cases.

Overall Feedback

This was the first time running a fully online January exam. It is clear that some questions worked better than others but overall we are happy that the exam measured the low-to-medium-level learning objectives, with the more high-level learning objectives being assessed in the summer exam.