UG Exam Performance Feedback
Second Year
2016/2017 Semester 2

COMP26120 Algorithms and Imperative Programming

David Rydeheard
Ian Pratt-Hartmann
Milan Mihajlovic

Comments

Question 1

This is a fairly standard question related to the examinable lab exercise on graph traversing algorithms. This question was taken by 186 out of 198 students. The overall feeling is that most of the students did quite well with the first two parts of the question, with some problems encountered in the final part related to algorithm complexity. The average mark was 14.6 with only a handful cases below 10 marks.

Question 2

This was actually a standard question on the Knapsack problem and its variants, slightly dressed up in an (actually not-so-fanciful) academic administration setting. The correct answers are pretty clear:

a) Greedy: this is fractional knapsack.
b) Dynamic programming: the matrix is small, so DP works well.
c) Branch and bound: the number of papers is still small.
d) Branch and bound (probably) again: same reason.

By and large, the question was done well. Most of the class demonstrated a good knowledge of the relevant algorithms, and knew which to apply when. Some students tended to give all the main algorithms (greedy, DP, B+B) but in the wrong order. In this case, I marked as flexibly as I could, to avoid the danger of Candidates who demonstrated knowledge of these algorithms ending up with nothing. Part d was intended to be more open-ended, and marks were given at least for a good attempt (of which there were some). The main trick was to see that there are interactions between the decisions you make for academics: submitting a paper for one academic A blocks its submission for another, B, and the next-best choice for B may block a paper for C. (In fact there is a rather clever polynomial-time way of solving this, but I did not expect Candidates to devise it.)

Otherwise, marks were lost because of the usual exam-technical issues: giving three (contradictory) answers instead of one; producingpseudocode that no human can read; trying to obfuscate details in the hope that the marker would not look too closely, and so on. But overall, the Examiners were happy with what the Candidates had learned.

Question 3

This question was related to the application of priority queues to task scheduling. This question was taken by 119 out of 198 students. Most of the students who took this question understood the problem well, selected the right data structure and found a suitable key to assign to the tasks that arrive to the priority queue. The main problem that students who didn't get it right encountered was not to select the right key. The average mark was 11.3, but the mark spread was fairly large.

Question 4

Almost all students took this question on graph traversal techniques and applications. Whilst some students performed very well with high-quality answers, the overall performance was poor, with an average of c50%. There were three reasons for this

- Clearly some students had not revised this material properly, nor attended the lectures or learnt from the podcasts.
- Many students did not read the question properly. The question is about directed graphs, where many students gave answers for trees only, and consequently lost marks.
- Many answers were too vague and imprecise to award marks.

The first part was to describe and program DFS, BFS and PS for graphs. Programs were mainly well presented, but explanations of the traversal techniques were poor on the whole, and rarely for graphs in general. Complexities: most students knew the answers, but explanations of why were not well handled on the whole.

The final part was an application: using DFS to create topological sortings of graphs. Students explained about acyclicity and some gave good proofs. Construction of topological sorts was attempted by many, but most did not observe that a DFS numbering, by itself, is not sufficient.