Algorithmics I

Exam Feedback December 2021

The following summarises how you coped with the exam questions. The solutions to the exam are also available on moodle.

Question 1.

On the whole this question was answered well. However, the majority of students left out the base case, even though there was the hint suggesting that you do not do this. Part (b) caused the most issues, with a number of students using the unit cost recurrent relation from the lectures.

The handout for section 2 includes the relevant details for this question and see the model solutions for the correct recurrence relation (with the base cases) and the correct table for part (b).

Question 2.

The first part of this question was in general answered very well. However, very few students understood what was being asked for in the second part of the question. The aim was for you to understand the trade offs between finding the mimum value in an array and a heap and updating an element in them (see the model solutions).

The handout and video for section 3 includes the relevant details for this question.

Question 3.

This question was answered very well. However a number of students left out any explanation for their answer which was required as part of the question.

The handout for section 4 includes the relevant details for this question.

Question 4.

This questions was on the whole answered well (particularly part (a)). Parts (b) and (c) were supposed to be more difficult and this proved to be the case (but most student got at least partial marks for each of these parts). For part (b) the main issue was correctly identifying the accepting states and considering the case when there are three or more c's in a row (see state q_3 in the model solutions). The same issues arose with part (c) when constructing the regular expression.

The handout for section 5 includes the relevant details for this question.

Learning Outcomes. The following maps the learning outcomes to the exam, assessed exercise and tutorial exercises.

1. Recognise, and be able to use, standard algorithmic design methods. Question 1, 2, assessed exercise and tutorial exercises

2. Apply the basic principles of algorithm analysis. Question 2, assessed exercise and tutorial exercises

4. Code standard efficient sorting algorithms. Tutorial exercises

5. Code fundamental graph algorithms - for search and traversal, shortest paths, minimum spanning trees, and topological sorting. Question 2

6. Describe classical algorithms for string searching, string comparison, and text compression. Questions 1, assessed exercise and tutorial exercises

7. Expound on the basic principles, and the practical implications of, the theory of NP-completeness. Question 3 and tutorial exercises

8. Follow NP-completeness proofs for particular problems. Question 3 and tutorial exercises

9. Deploy various strategies for dealing with computational problems that are apparently intractable. Tutorial exercises

10. Provide examples of the computability and unsolvability, and know some standard examples of unsolvable problems. Question 4 and tutorial exercises