Lecture 12
Semester 2 Revision

COMP26120

Giles Reger, Milan Mihajlovic and Lucas Cordeiro

May 2019

Headlines

May Exam is 2 hours

There will be three questions, you must answer all three

There will be three sections:

- Section A: Graphs and Graph Algorithms [20 marks]
- Section B: Complexity and NP-Completeness [10 marks]
- Section C: Algorithmic Tech. and Number Theoretic Alg. [30 marks]

Section C may refer to/rely on topics covered in Semester 1 related to complexity analysis and data structures.

Reminder: Always read the question and take note of the marks available.

Reminder: Course Assessment

Coursework (50%):

- 25% each semester
- In each semester
  - 20% from coding exercises
  - 5% from 3 online quizzes (see website for when)
  - Quizzes provide early feedback on conceptual issues

Exam (50%):

- 15% first semester, 35% second semester
- Material (general concepts) covered in labs is examinable
- Note: new staff means exams likely to be a little different in style

Reminder: Course Assessment

Coursework (50%): ← note split

- 25% each semester
- In each semester
  - 20% from coding exercises
  - 5% from 3 online quizzes (see website for when)
  - Quizzes provide early feedback on conceptual issues

Exam (50%): ← note split

- 15% first semester, 35% second semester
- Material (general concepts) covered in labs is examinable
- Note: new staff means exams likely to be a little different in style

Examinable Topics

Graphs and Graph Algorithms

- Graph representation
- Graph traversal (depth-first/breadth-first)
- Transitive closure (Floyd-Warshall)
- Shortest path (Dijkstra, Bellman-Ford)

Complexity and NP-Completeness

- Concepts of P, NP, NP-hard, NP-complete
- Proofs of NP-completeness via reduction
- Recalling well-known NP-complete problems

Examinable Topics

Algorithmic Techniques

- Being able to describe and apply
  - Divide and conquer
  - Branch and bound
  - Greedy algorithms
  - Dynamic programming
  - Linear programming
- Read a problem statement, identify which technique to apply, and design an appropriate solution (using appropriate data structures)
- Analysing the complexity of your solution

Number Theoretic Algorithms

- Recalling facts about prime numbers
- El Gamal Encryption
- Primality testing

Relation to Previous Papers

Linear Programming is included in this year's syllabus and does not appear in previous papers.

Previous years had two examinable labs (Dijkstra's and Knapsack) that would have an associated question in the exam. This is no longer the case. You will notice these topics coming up a lot in previous years for this reason; you should not assume this trend to continue.

With respect to past papers:

- 17/18 - questions 1 and 2 in paper 2
- 16/17 - questions 1 and 2 in paper 1, questions 1, 2, 4 in paper 2
- 15/16 - questions 1, 2, parts of 4 (d,e) in paper 2
- 14/15 - questions 1and 2c in paper 1, questions 1, 2, 4 in paper 2

Example Question

You are playing a strategy fighting game and want to maximise the damage done by your fighting force. The rules of the game are as follows:

- You can create two kinds of units: a solider and a tank
- A solider does 3 damage and a tank does 2 damage
- You can have at most 100 units
- You need enough transport for your units. You have one transporter that can carry 20 soldiers. Your tanks come with transporters for free. Each tank transporter carries 3 tanks and has an extra seat for 1 soldier if required e.g. for every 3 tanks you can transport 1 soldier.

Cast this problem as an optimisation problem to optimise the amount of damage. Use the Simplex algorithm to find the number of soldiers and tanks you need and how much damage you can do. As a simplification, assume that you can have a fractional part of a unit. Comment on why this is an important simplification and what the implications of not making this simplification would be. (8 marks)

Summary

You should answer all questions in the exam.

Read the question (and the marks)!

Good luck!