Lecture 21: Summary / Preparing for the Exam

COMS10007 - Algorithms

Dr. Christian Konrad

07.05.2019

Outlook

Long-term Goal:

Ability to design and analyze algorithms

1. **COMS10007: Algorithms** (1st year)
2. COMS21103: Data Structures and Algorithms (2nd year)
3. COMS31900: Advanced Algorithms (3rd year)
4. ... (4th year)

Projects:

- Final projects
- Summer internships (after the second year)
- PhD theses

Exam

- June, 6th at 9:30 am
- 2 hours

The Exam will test both your **skills** and **knowledge**

Two Key Ingredients:

1. *Tools/skills:* O-notation, recurrences, loop invariants (induction), mathematics (e.g., bounding sums), ...
2. *Knowledge:* algorithms, algorithmic design principles (divide-and-conquer, dynamic programming, ...)

Exam

Key Skills:

- O-notation (Ω, Θ), formal proofs (e.g. is $f \in O(g)$?), racetrack principle, ...
- Recurrences, substitution method, recursion tree method, ...
- Runtime analysis

Key Knowledge:

- Precise definitions, principles (e.g., dynamic programming)
- Algorithms (e.g. sorting, binary search, Fibonacci, ...)

Material: Everything, except excluded material

Week	Day	Date	Topics
13	Mon	28-Jan	Lecture 1: Introduction, peak finding
	Tue	29-Jan	Lecture 2: O-notation
			Remarks: Throughout this course, log(n) denotes the binary logarithm, i.e., log(n) = $\log_2(n)$. In the slides used in th of log(n) was wrongly computed with 1/n, while it should of course be 1/(n ln(2)), with ln(n) being the logarithm to in the slides.
14	Mon	04-Feb	Lecture 3: Theta, Omega, RAM Model
	Tue	05-Feb	Lecture 4: Linear and binary search, proofs by induction
	Tue	05-Feb	**Exercise class 1**
			Remarks: Exercises on O-notation, Omega, and Theta can all be solved by finding constants c, n_0 such that the defin Theta, Omega) is fulfilled. Recall that any constants that fulfill the respective definition are fine, i.e., you do not nee ones. Solutions will be provided soon.
15	Mon	11-Feb	Lecture 5: Loop invariants, insertion sort
	Tue	12-Feb	Lecture 6: Sorting problem, first part of merge sort algorithm
16	Mon	18-Feb	Lecture 7: Second part of merge sort, maximum subarray problem (slides of lectures 6 and 7 are
	Tue	19-Feb	Lecture 8: Trees, first part of heap-sort
	Tue	19-Feb	**Exercise class 2**
17	Mon	25-Feb	Lecture 9: Second part of heap-sort (slides of lectures 8 and 9 are combined)
	Tue	26-Feb	Lecture 10: Quicksort
18			**Reading Week**
19	Mon	11-Mar	Lecture 11: Runtime of Quicksort
	Tue	12-Mar	Lecture 12: Sorting LB, Countingsort, Radixsort
	Tue	12-Mar	**In-class Test**
20	Mon	18-Mar	Lecture 13: Recurrences (substitution method, recursion-tree method)
	Tue	19-Mar	Lecture 14: Recurrences continued (slides of lectures 13 and 14 are combined)
21	Mon	25-Mar	Lecture 15: Fibonacci Numbers
	Tue	26-Mar	Lecture 16: Dynamic Programming - Pole Cutting
	Tue	26-Mar	**Exercise class 3**
22	Mon	01-Apr	Lecture 17: Dynamic Programming - Matrix Chain Parenthesization
	Tue	02-Apr	Lecture 18: Elements of Dynamic Programming 1
			Easter Break
23	Mon	29-Apr	Lecture 19: Elements of Dynamic Programming 2

Material

O-notation

$\downarrow$

Loop Invariants

$\downarrow$

Sorting

$\downarrow$

Recurrences (Divide-and-Conquer)

$\downarrow$

Dynamic Programming

Excluded Material

No Questions on: (only holds for exam, not for repeat exam!)

- RAM Model
- 2D peak finding
- Maximum Subarray
- Exercise sheet 4
- No loop invariant...!

(However still helpful to know!)

How to study for the Exam?

Overview

- Get a complete picture of the material
- Story behind it

Algorithms

- Understand algorithms, explain in words
- Example runs
- Understand their runtimes
- No need to give code

How to study for the Exam?

Answer the questions: Knowledge

- What is ... (any key word/concept/algorithm)?

  E.g. Θ notation? a binary tree? a divide-and-conquer algorithm? the substitution method? the sorting problem? binary search? the best-case runtime of merge sort? ...

- How does ... work? (e.g. any algorithm, design principle, concepts, ...)

- What is the definition of...?

How to study for the Exam?

Exercise sheets:

- Problem solving
- Exam questions similar (slightly easier, fewer technicalities)

Concepts:

- Know what to do (O-notation, recurrences, etc.)
- Algorithm versus concept

Mock Exam:

- Will be put online later this week
- Attention: Tests only a subset of what is relevant

First Edition

Things I was very pleased with:

- Attendence
- Office hours, drop-ins
- In-class test

Discussion:

- More/longer exercise classes
- More exercises of different difficulties
- Practical relevance, toy problems