

COMMONWEALTH OF AUSTRALIA

Copyright Regulations 1969

This material has been reproduced and communicated to you by or on behalf of Monash University pursuant to Part VB of the Copyright Act 1968 (the Act). The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act. Do not remove this notice

Prepared by: [Arun Konagurthu]

FIT3155 S2/2018:
Advanced Data Structures and Algorithms
Unit Synopsis

Faculty of Information Technology, Monash University

What's this course about?

- The subject **is** about **efficient problem-solving** with computers.
- The subject **is** about space and time efficient data structures and algorithms.
- The subject **is not** (mainly) about **programming**.
- The subject just happens to use **Python** as the programming language in which lab work (etc.) is done. This subject is really **language agnostic**.
- Algorithms in this unit will be presented/described in English, pseudo-code, procedural set of instructions, as convenient.

Coverage

The subject aims to cover:

- 1 Advanced algorithmic problem-solving strategies, e.g. essential techniques such as

- ▶ linear programming,
- ▶ randomised algorithms
- ▶ approximation algorithms,
- ▶ etc.

advanced analysis such as

- ▶ amortised analysis,
- ▶ space- and time-efficient of data structures,
- ▶ estimation of space and time complexity of algorithms,
- ▶ etc.

Unit Schedule (**Tentative!**)

Week 1

Linear-time exact pattern matching

Week 2

Advanced string data structures.

Week 3

Disjoint set data structure etc.

Week 4

Amortised analysis

Week 5

Adv tree data structures-1

Week 6

Adv tree data structures-2

...continues in next slide

Unit Schedule (**Tentative**) ...contd.

Week 7

Compression-based algorithms

Week 8

Number-theoretic algorithms

Week 9

Linear programming

Week 10

Advanced graph algorithms

Week 11

Approximation algorithms

Week 12

Compact data structures

Course material

- Your main portal will be, as you already know, the unit's **Moodle** page.
- Material available on moodle will include:
 - ▶ Introductory Notes
 - ▶ Lecture slides
 - ▶ Additional references
 - ▶ Practical (Lab) sheets
 - ▶ Tutorial sheets
- **Remember(!)** to set your e-mail forward.

Main textbook references

- ① Cormen et al., Introduction to Algorithms
 - ▶ Chapters: 18, 20, 29, 31, 33 and 35.
- ② Weiss, Data structures and Algorithm analysis
 - ▶ Chapters: 7, 8, 11 and 12.
- ③ Gusfield, Algorithms on Strings, Trees, and Sequences
 - ▶ Chapters: 1, 2, and 6.

This subject will aid your development into professional programmers, technicians, software engineers, computer scientists, etc. Therefore, read beyond what is prescribed for this unit.

Course structure

Lectures: (**refer to your timetable for time/venue**)

- 2 hours/week – refer to your timetable for time/venue

Tutorials/Tutes (**refer to your timetable for time/venue**)

- 1 hour fortnightly, **starting Week 2**.

Practicals/Pracs/Lab (**refer your class timetable for time/venue**)]

- 2 hour/week, **starting Week 2**

Assignments

- 2 assignments due over the teaching period
- Assignment 1 due end of week 8
(released end of week 5)
- Assignment 2 due end of week 12
(released end of week 9)

Marks and Hurdles – IMPORTANT

To pass FIT2004

- Your marks must average to **at least** 50% of the total marks for this unit, **and**
- you must pass each of the hurdle (see next slide).

Assessment – IMPORTANT

In-semester assessment contributes to 40% of total unit marks

- Assessed Prac marks = $\frac{40}{100}$, with 20 marks per each of the 2 assignments.
- **HURDLE 1:** You should score **at least** 40% of the in-semester marks ($\geq \frac{16}{40}$ marks).

End-of-Semester Examination = 60% of total semester marks

- End-of-Semester (written) exam carries 60 marks.
- **HURDLE 2:** You should score **at least** 40% of the exam marks ($\geq \frac{24}{60}$ marks).

Student responsibilities

Responsibilities

- Regularly attend the weekly activities.
- Diligent self-study (nominally 7 hours/week).
- This subject is best understood practically. So, assimilating various concepts and practicing them (**a lot!**) is the key to success.

Etiquette

- No noise and distractions during the lectures
- Use rear door if you must arrive late or leave early.
- Turn off your mobile phones.

Plagiarism and Cheating

- Monash University takes plagiarism and cheating very seriously. There are **severe penalties** for them.
- Read Monash University's [\[Plagiarism and cheating Policy\]](#)
- In a nutshell, Plagiarism is legitimately using someone else's work, but not acknowledging it. Cheating is pretending that someone else's work is your own, in order to gain an unfair advantage.
- It is **OKAY** to work together in trying to understand concepts. **Peer Assisted learning** (PAL) is fun and a great way to **cross-fertilize** each others' thinking – but each student must be *conscientious* in her/his work, write code for the entire assignments alone, and be able to explain and modify it on request
- **All scripts will be screened through a powerful software for detecting software plagiarism.**

Language and Learning Officer

Mr Noriaki Sato

- noriaki.sato@monash.edu
- 9905 1775

Individual/group consultations and courses

Hargrave-Andrew Library

- <http://www.lib.monash.edu/contacts/learning-skills.html>
- <http://www.monash.edu.au/lls/llonline>

--o0o--

END

--o0o--