Algorithms and Data Structures

COMP261
Course Overview

Alex Potanin, Yi Mei

yi.mei@ecs.vuw.ac.nz

The Team

- **Coordinator**:
  - Yi Mei: yi.mei@ecs.vuw.ac.nz
- **Lecturers**:
  - Yi Mei (First half): yi.mei@ecs.vuw.ac.nz
  - Lindsay Groves (Second half): lindsay@ecs.vuw.ac.nz
- **Senior Tutor**
  - Morgan Atkins: morgan.atkins@ecs.vuw.ac.nz
- **Tutors**:
  - Damien O'Neill
  - Alice Little
  - Christopher Parry
  - Balgmi Nam
  - Daniel Ko
  - Hrshikesh Arora
  - Joshua Weir
  - James Watt
  - Thomas Herdson
  - Inti Resende

Lectures, Tutorials, Helpdesks

- Lectures: Monday 10:00-10:50, HMLT205
- Lectures: Tuesday 12:00-12:50, MCLT101
- Lectures/Tutorials/No lecture: Thursday 12:00-12:50, MCLT101

- Tutorials
  - Start from **week 2**
  - 10 slots, see timetable in the course homepage
  - Signup through MyAllocator
  - Small groups (~30), go through (tough parts of) the lecture content
  - Lectures cannot cover every single detail
  - Tutorials can be very helpful

Lectures, Tutorials, Helpdesks

- Helpdesks:
  - From **Week 2**
  - CO 242b
  - **15:00-16:00 Monday to Friday**
  - **14:00-16:00 (2 hours) for the weeks when the assignments are due**.
- Helpdesks tutors are on the course homepage.
- First week: no helpdesks, but will be a tutorial in lecture slot
- No scheduled labs

Course Objectives

- Select, adapt, and implement a wide range of standard algorithms and data structures to construct software solutions to complex problems
- Understand algorithms described in pseudocode and to use analysis of an algorithm's time and space requirements to determine applicability to a problem
- Recognise the distinction between "easy" problems, 'hard' (NP) problems, and uncomputable problems and the consequences for constructing algorithms and programs for such problems
- Know and be able to implement important algorithms related to graphs, searching, parsing, basic graphics rendering, and B-Trees

Course Materials

- No compulsory “text book”, but below will be useful
  - **Book: “Algorithms and Data Structures”** – a selection of chapters from various textbooks compiled by Alex Potanin, Pearson (some copies may be around, especially second hand, borrow from library)
  - **Wikipedia** pages: extremely good resource on algorithms.
  - **Lecture notes** on the course website: https://ecs.victoria.ac.nz/Courses/COMP261_2018T2/

- Tutorials and helpdesks will be very helpful

Assessments

- Tests (20%) and Final exam (50%)
  - Mid-term test: 45 mins, in lecture (across two theatres)
  - Exam: exam period
- Assignments (30%)
  - 5 assignments, roughly every 2-3 weeks
  - 6% each
  - Deadlines:
    - Due at **23:59 Sundays**
    - 20 marks off for first 24 hours late, 40 marks off for next 24 hours, 0 marks more than 2 days late.
    - 3 "late days" for the whole course, so use wisely
    - Further extensions is only allowed under special circumstances (having other assignments due at the same time is not special)
    - **IN PERSON MARKING!!! 10%-100% PENTALTY IF YOU MISS IT!**

- **To pass, you have to achieve at least 40% in the final exam**

Assignments

- Assig 1: Graph basics
- Assig 2: Pathfinding and articulation points
- Assig 3: 3D graphics
- Assig 4: Parsing
- Assig 5: Indexing, B+ trees, low-level file structures

- Submissions
  - **Code + report** (Submit through the system)
  - Signup a 15-minute session for marking, demonstrate program and explain key details to the marker (tutor)
  - Marked in the two weeks after the due date (e.g. week 5-6 for A1)
  - Marking done in CO241
  - Signup sheets will be posted outside CO241

Workload

- 10 hours per week
  - Lectures and tutorials: 3 hours
  - Review and reading: 2 hour
  - Assignment: 5 hours

- Assumptions:
  - **COMP103**: collections, complexity (Big-O), searching and sorting
  - **Math161/ENGR123**
  - Basic 2D geometry
  - Ability to find things out by yourself (COMP261 is not spoon-feed)

- Start the assignment as early as possible!!!

Rules and Policies

- Plagiarism is **NOT tolerated**
  - Submit someone else’s work as your own, including
    - material from a published source such as a library book, a journal article, etc.
    - material from an on-line software library, web pages, etc.
    - the work of another student, friend, relative, etc.
- Penalties:
  - zero marks for the work to which it relates
  - greater penalties in accordance with the University's Statute on Student Conduct
- If you discussed with other students, state it clearly in the report to avoid plagiarism issue
- https://ecs.victoria.ac.nz/Main/ComputerSciencePlagiarism

Class Representative

- A class rep is the bridge between the lecturer and the students. They are not meant to be a note taker or class life coach, but instead to facilitate feedback by communicating regularly with the class and the course coordinator.

- Introductory Video here:
  - https://www.youtube.com/watch?v=ofRy3oIoXD4
- Register here:
  - https://www.vuwsa.org.nz/class-representatives/

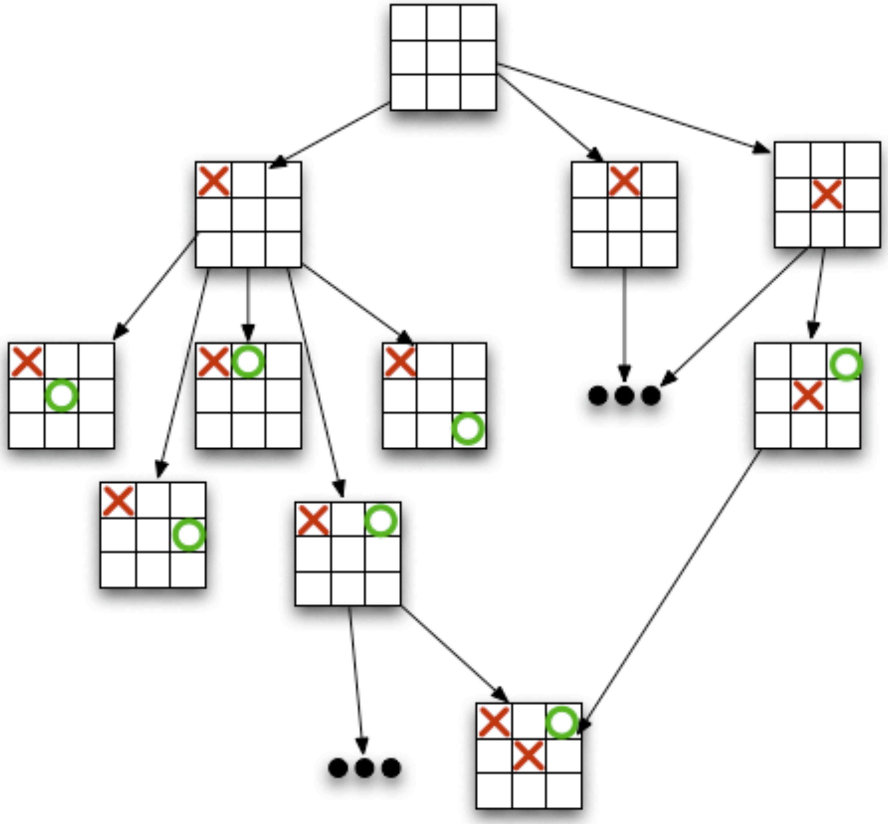
Graph

- Many real-world applications
  - places with connections

    airports & flights,
    intersections & roads,
    network switches and cables
    ….
  - entities with relationships

    social networks,
    biological models
    web pages ….
  - states and actions

    games, plans, …..



Graph

- A collection of **nodes**
- A collection of **edges** (directed and undirected)
  - We only consider directed edges
  - Undirected edge can be seen as a pair of directed edges
  - (A, B) can be seen as (A -> B) and (B -> A)
- Relationship between nodes and edges
  - Nodes form edges
  - Edges connect nodes

- Other properties in graph
  - Multi-graph: multiple edges between the same pair of nodes
  - Loops

- Other complex properties
  - No turn right/left