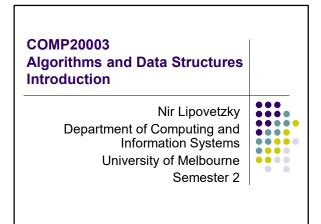
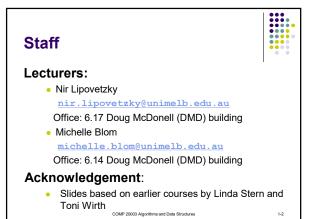
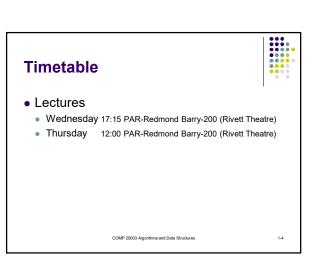
# Introduction

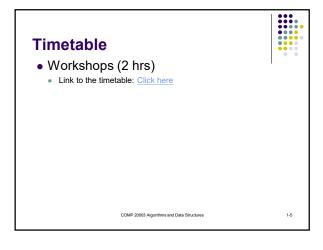
# The University of Melbourne COMP20003 Algorithms and Data Structures

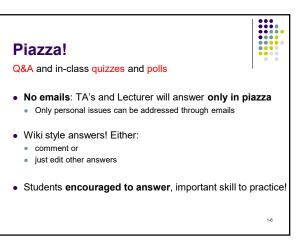












# The University of Melbourne COMP20003 Algorithms and Data Structures

# What you will learn in this subject – and why.



- · A number of useful algorithms.
- · How to analyze algorithms for efficiency.
- Build further proficiency in C programming through implementing algorithms.

1-7

#### **Outline of the first few lectures**



- Algorithms: general
- This subject: details
- Algorithm efficiency
- Computational complexity
- Data structures
  - Basic data structures
  - · Algorithms on basic data structures
  - · Complexity analysis of basic algorithms

COMP 20003 Algorithms and Data Structures

1.8

#### What is an algorithm?



COMP 20003 Algorithms and Data Structure

### **Algorithms**

- Al Khwarizmi
  - Baghdad, 9<sup>th</sup> century
  - Textbook:
    - Arabic numerals
    - decimal positional number system
    - how to add
    - multiply
    - extract square roots
    - calculate pi COMP 20003 Algorithms and Data Structures

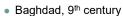
#### **Algorithms**

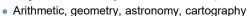
- Al Khwarizmi
  - Baghdad, 9<sup>th</sup> century
  - Also
    - Showed how to solve linear and quadratic equations.
    - Corrected Ptolemy's estimate for size of Mediterranean.
    - Analyzed Hebrew calendar 19-year cycle.
    - and much more!

COMP 20003 Algorithms and Data Structures

### **Algorithms**







- Leonardo Pisano Bigollo, aka Leonardo di Pisa, aka Fibonacci (filius Bonacci)
  - Italy, 13<sup>th</sup> century
  - Brought Arabic numerals to the west
  - Popularized the Fibonacci number series

COMP 20003 Algorithms and Data Structures

1-12





# The University of Melbourne COMP20003 Algorithms and Data Structures

# Algorithm classification Classified by task:

- Sorting
- Searching
- Numeric
- Routing
- Scheduling
- etc.

#### Algorithm classification II



- · Classified by approach:
  - Brute force
  - Divide and conquer
  - Decrease and conquer
  - Greedy
  - etc.

### Algorithm classification III



- · Classification based on the answer:
  - Exact
  - Approximation
  - Heuristic

### General approach in this subject



- Introduction to data structures, algorithms, and computational complexity.
- For every algorithm:
  - How it works
  - Complexity analysis
  - Implementation
- NP-completeness

#### Algorithms in the real world 20th century



- · Navigation software: get shortest path to destination.
  - -- And do it quickly.
- · Connect towns or houses to telecommunications network.
  - -- With the least cost in wire.

### Algorithms in the real world: 21st Century



- Self-Driving cars: trajectory planning, object recognition, Localization etc.
- Determine whether someone should get a mortgage. Determine how likely he's to pay back?

# The University of Melbourne COMP20003 Algorithms and Data Structures

#### Outline of the first few lectures



- Algorithms: general
- → This subject: details
  - Algorithm efficiency
  - Computational complexity
  - Data structures
    - Basic data structures
    - · Algorithms on basic data structures
    - Complexity analysis of algorithms on basic ds's

### This subject: some details



- Lectures:
  - Theory of algorithms
  - High-level how-to of algorithms
  - A little bit of code
- Workshops: tutorial + computer lab
  - Apply theory
  - Practice implementing

#### Workshops and assignments



- C programming in the workshops and for assignment submissions can be done on the platform of your choice, BUT...
- · We are supporting one platform and compiler
  - MobaXterm: for ssh, also has an editor
    - Atom: alternative opensource editor with 1000s of extensions
  - CIS machines new virtual machines, to test code:
    - nutmeg.eng.unimelb.edu.au
    - dimefox.eng.unimelb.edu.au

C compiler: gcc
COMP 20003 Algorithms and Data Structures

# Workshops and assignments



- See document on LMS:
  - Resources ->Introduction to UNIX (and MobaXterm)
- Note, however, new machines...
- See more documentation on the LMS:
  - Resources->The New CIS Virtual Machines

### **Workshops and Assignments**



- MobaXterm / atom:
  - Installed on laboratory machines
  - Download (free) for home use:
  - http://mobaxterm.mobatek.net/download.html or http://atom.io
  - For problems: see your tutor
- · CIS (new) virtual machines:
  - Red Hat Enterprise Linux 6.5 dimefox.eng.unimelb.edu.au
    - nutmeg.eng.unimelb.edu.au
  - eng.unimelb.edu.au/itservices/students/general\_unix.html For problems: lodge a ticket at

, select "Teaching Support", start your request with "New CIS Teaching Servers'

# **Workshops and Assignments**



- Working from home...
  - You must connect via the university's VPN.
  - · Direct access from the Internet is not permitted
- VPN:
  - https://studentit.unimelb.edu.au/findconnect/vpn
  - You will have to install Cisco AnyConnect
  - There is a web launcher on this page.
  - If the web launcher doesn't work, there are instructions for manual install.
  - For problems:

  - or ring 8344 0888 M-F 8AM-6PM
- More help...

# Introduction

# The University of Melbourne COMP20003 Algorithms and Data Structures

#### More help



- Student IT Support: web requests, drop-in centers and hours:
  - http://studentit.unimelb.edu.au/contact/index.html
- It is strongly suggested that you work out your machine access this week.

COMP 20003 Algorithms and Data Structur

ta Structures

#### **Books**



- Prescribed textbook:
  - Steven Skiena, The Algorithm Design Manual.
  - Available as an eBook from the MU library.
  - <a href="http://library.unimelb.edu.au/">http://library.unimelb.edu.au/</a> → Catalogue → eBooks → Skiena
  - Note: The copyright license does permit you to download and print for your own personal study.
- Other highly recommended books on reserve (ERC High use area):
  - Sedgewick, Algorithms in C vol 1, and Algorithms in C, Part 5: Graphs
  - Levitin, Introduction to the Design and Analysis of Algorithms.
  - · Cormen, Leiserson, and Rivest, Algorithms.

#### **Assessment**



- Continuous assessment 30% (week 4 and 10)
  - Two C coding assignments + experimentation.
  - Hurdle 15/30.
- Mid-semester test 10% (week 7)
  - 6th of September, 12 1pm, Wilson Hall
- Final examination 60%
  - 3 hours
  - The practical component will *not* be at a computer.
  - Hurdle test+final exam 35/70

1-27