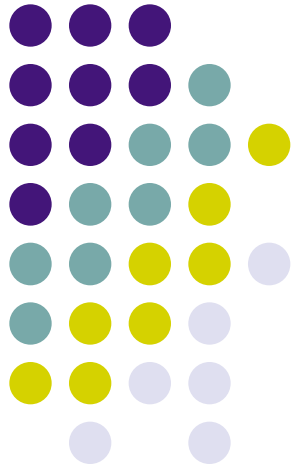
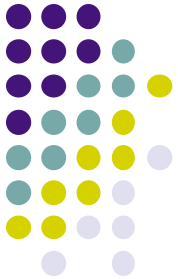
COMP20003 Algorithms and Data Structures Introduction

Nir Lipovetzky
Department of Computing and Information Systems
University of Melbourne
Semester 2 2016





Staff

Lecturer:

- Nir Lipovetzky

  nir.lipovetzky@unimelb.edu.au

Office: 6.17 Doug McDonell (DMD) building

Acknowledgement:

- Slides based on earlier courses by Linda Stern and Toni Wirth

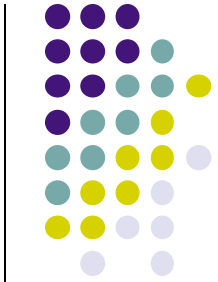
Staff

- **Tutors:**
  - Head Tutor: Grady Fitzpatrick

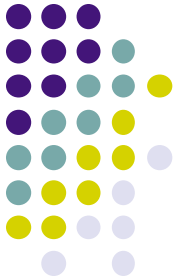
    `grady.fitzpatrick@unimelb.edu.au`

  - Angus White
  - Anh Vo
  - Chung Man Lam (Raymond)
  - Curtis Musgrave-Evans
  - Aidan Dang
  - Wenxi Wang

Timetable



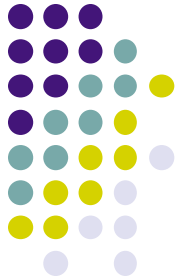
- Lectures
  - Thursday 12:00 Charles Pearson Theatre (ERC)
  - Friday 17:15 Carrillo Gantner Theatre (Sidney Myer Asia)



Timetable

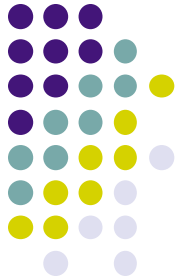
- Workshops (2 hrs)
  - Monday 9:00 Alice Hoy 210
  - Monday 9:00 Doug McDonell 502
  - Monday 13:00 Doug McDonell 502
  - Monday 19:15 Alice Hoy 109
  - Tuesday 15:15 Doug McDonell 502
  - Tuesday 19:15 Alice Hoy 108
  - Tuesday 19:15 Alice Hoy 109
  - Tuesday 19:15 Alice Hoy 222
  - Wednesday 19:15 Alice Hoy 108
  - Thursday 9:00 Doug McDonell 502
  - Friday 19:15 Alice Hoy 108

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



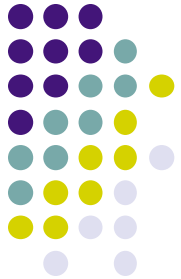
- A number of useful algorithms.
  - Having a library of algorithms at your fingertips helps you solve new problems.
- How to analyze algorithms for efficiency.
  - Gives you the ability to choose the best algorithm for the task at hand.
  - Gives you ability to analyze new algorithms.
- Build further proficiency in C programming through implementing algorithms.
  - Strong foundation for problem solving and programming in any language.

Outline of the first few lectures

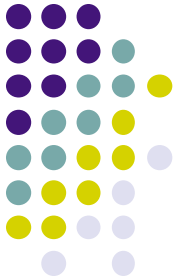


- Algorithms: general
- This subject: details
- Algorithm efficiency
- Computational complexity
- Data structures
  - Basic data structures
  - Algorithms on basic data structures
  - Complexity analysis of basic algorithms

What is an algorithm?

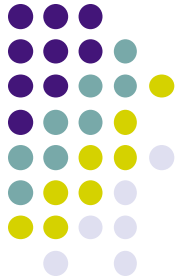


What is an algorithm?



- A set of steps to accomplish a task:
  - A cooking recipe.
  - A procedure for doing laundry.
  - A procedure for getting dressed.
  - A procedure for diagnosing disease.
  - A procedure for applying for a Masters.
  - *etc.*

computer
What is a^ algorithm?



- An algorithm with the following properties:
  - Precisely defined:
    - GPS route: if traffic is “bad” doesn’t work.
  - Defined input.
  - Defined output.
  - Correct.
    - Exactly correct, or correct to within ε.
  - Terminates within a reasonable period of time.

Algorithms

- Al Khwarizmi
  - Baghdad, 9th century
  - Textbook:
    - Arabic numerals
    - decimal positional number system
    - how to add
    - multiply
    - extract square roots
    - calculate pi

Cartoon from Sydney Harris

Algorithms

Hunter Johnson: Creative Commons
http://upload.wikimedia.org/wikipedia/commons//04/Al-Khwarizmi%2C_Khiva.jpg

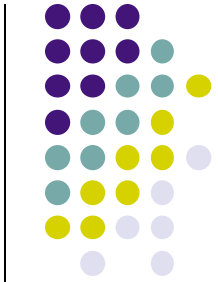
- Al Khwarizmi
  - Baghdad, 9th century
  - Also:
    - Showed how to solve linear and quadratic equations.
    - Corrected Ptolemy's estimate for size of Mediterranean.
    - Analyzed Hebrew calendar 19-year cycle.
    - and much more!

Algorithms

Image in Public Domain

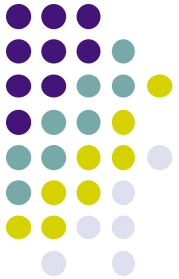
- Al Khwarizmi
  - Baghdad, 9th century
  - Arithmetic, geometry, astronomy, cartography
- Leonardo Pisano Bigollo, aka Leonardo di Pisa, aka Fibonacci (filius Bonacci)
  - Italy, 13th century
  - Brought Arabic numerals to the west
  - Popularized the Fibonacci number series

Algorithm classification



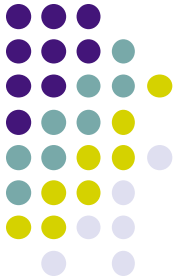
- Classified by task:
  - Sorting
  - Searching
  - Numeric
  - Routing
  - Scheduling
  - *etc.*

Algorithm classification II



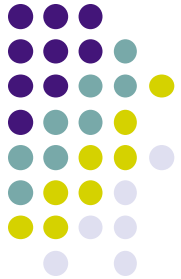
- Classified by approach:
  - Brute force
  - Divide and conquer
  - Decrease and conquer
  - Greedy
  - *etc.*

Algorithm classification III



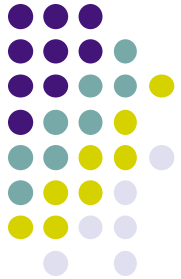
- Classification based on the answer:
  - Exact
  - Approximation
  - Heuristic

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



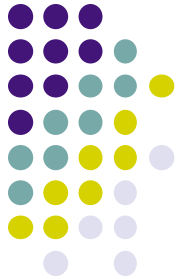
- Navigation software: get shortest path to destination.

  -- And do it quickly.

- Connect towns or houses to telecommunications network.

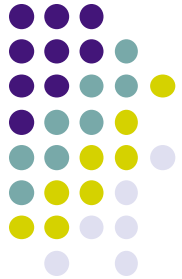
  -- With the least cost in wire.

Algorithms in the real world II



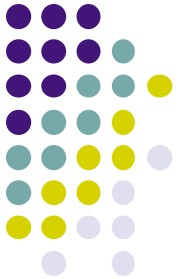
- Given a set of subjects and prerequisites, determine the minimum number of semesters remaining to complete degree.
- Determine whether a newly sequenced gene is similar to anything already known gene. Determine how similar?

Outline of the first few lectures



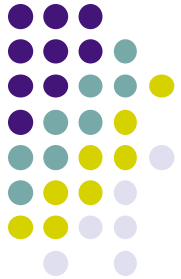
- Algorithms: general
- This subject: details
- Algorithm efficiency
- Computational complexity
- Data structures
  - Basic data structures
  - Algorithms on basic data structures
  - Complexity analysis of algorithms on basic ds's

This subject: some details



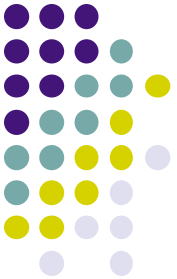
- Lectures:
  - Theory of algorithms
  - High-level how-to of algorithms
  - A little bit of code
- Workshops: tutorial + computer lab
  - Apply theory
  - Practice implementing

Workshops and assignments



- C programming in the workshops and for assignment submissions can be done on the platform of your choice, BUT…
- We are only supporting one platform and one set of tools.
  - MobaXterm: for ssh, also has an editor
  - CIS machines new virtual machines:
    - nutmeg.eng.unimelb.edu.au
    - dimefox.eng.unimelb.edu.au
  - C compiler: gcc

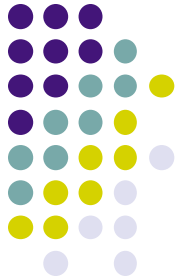
Workshops and assignments



- See document on LMS:
  - *Resources→Introduction to UNIX (and MobaXterm)*

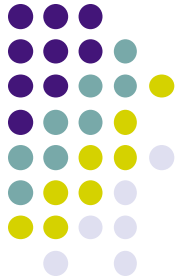
- Note, however, new machines…
- See more documentation on the LMS:
  - *Resources->The New CIS Virtual Machines*

Workshops and Assignments



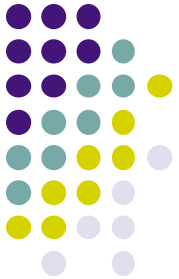
- MobaXterm:
  - Installed on laboratory machines
  - Download (free) for home use: http://mobaxterm.mobatek.net/download.html
  - For problems: see your tutor
- CIS (new) virtual machines:
  - Red Hat Enterprise Linux 6.5
    - dimefox.eng.unimelb.edu.au
    - nutmeg.eng.unimelb.edu.au
  - http://ithelp.eng.unimelb.edu.au/student/general_unix.html
  - For problems: lodge a ticket at http://ithelp.eng.unimelb.edu.au/servicedesk/, select “Teaching Support”, start your request with “New CIS Teaching Servers”
- Working from home…

Workshops and Assignments



- Working from home…
  - You must connect via the university's VPN.
  - Direct access from the Internet is not permitted.
- VPN:
  - https://its.unimelb.edu.au/help/networks-access/networks-internet/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:
    - Lodge a help request at this URL
    - or ring 8344 0888 M-F 8AM-6PM
- More help…

More help



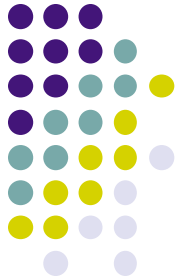
- Student IT Support:

  web requests, drop-in centers and hours:

  - http://studentit.unimelb.edu.au/contact/index.html

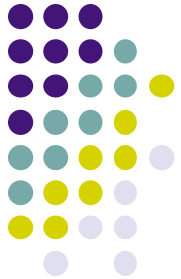
- It is strongly suggested that you work out your machine access this week.

Books



- Prescribed textbook:
  - Steven Skiena, *The Algorithm Design Manual.*
  - Available as an eBook from the MU library.
  - http://library.unimelb.edu.au/ → 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%
  - Two C coding assignments + experimentation.
  - Hurdle 15/30.
- Mid-semester test 10%
- Final examination 60%
  - 3 hours
  - The practical component will *not* be at a computer.
  - Hurdle test+final exam 35/70