COMP1005
Programming and Algorithms

Module Introduction

Jamie Twycross

Overview

- Lecturer introduction
- Module introduction:
  - aims and objectives
  - resources
  - lectures, labs and tutorials
  - assessment
  - course books
  - software

Lecturer Introduction

- Dr Jamie Twycross, Assistant Professor
  - B.Sc. Mathematical Physics (Imperial College, London)
  - M.Sc. Evolutionary & Adaptive Systems (University of Sussex)
  - Ph.D. Computer Science (University of Nottingham)
- Worked in industry - HP Research Labs
- Been at Nottingham since 2004
- Member of the Intelligent Modelling & Analysis Group
- Research centres around Computational Biology:
  - Machine Learning/Artificial Intelligence
  - Data Analytics/Visualisation
  - Computational and Mathematical Modelling
- Also interested in Security, Hardware, Blockchain, ...
- Teach 1^{st} year PGA module, supervise 2^{nd} year group projects, 3^{rd} year dissertations and Masters projects
- Computer Science Examinations Officer

Module Introduction

Aims and Objectives

- Aim:
  - introduce basic principles of programming and algorithms
- Objectives:
  - teach **fundamental programming constructs**, types and variables, expressions, control structures and functions in C
  - teach how to design and analyse simple **algorithms and data structures**, allowing efficient storage and manipulation of data
  - familiarise students with **software development methods**, including documentation, testing, debugging, and the use of software tools, such as version control systems and continuous integration

Resources

- Module **Moodle** page:

  Programming and Algorithms
  (COMP1005 UNUK) (AUT1 22-23)

- Module **Office Hours**:
  - Tuesday 11:00-12:00 in-person B48
- Moodle **Announcements**
- **GitLab** for coursework - **projects.cs.nott.ac.uk**

Lectures

- Lectures are combination of **in person lectures** and **pre-recorded videos**
- **In person lecture** every **Tuesday 10:00-11:00** in B52 Business School South
- **Online lecture engagement** every **Monday 10:00-12:00** - review and study weekly lecture material
- Weekly **lecture package** released on **Moodle** every **Monday**:
  - slides
  - videos discussing slides
  - quizzes
  - suggested reading
  - recording of in person lecture
- **Take notes** - I say more in the videos than the words on the slides!
- Lectures are **sign-posts** to what you need to study and practice - you need to **study outside of lectures**

Labs

- Labs are **face-to-face**
- Set a number of **practical programming problems**:
  - material generally available before lab
- One **two hour lab** per week
- Labs shared with COMP1006 and COMP1007
- No COMP1005 labs weeks 4, 8 and 11
- Your cluster is assigned a lab slot
- Lecturers and teaching assistants present
- You **learn programming by doing it**
- You will need to **program outside of labs**
- Labs start this week

Tutorials

- Tutorials are **face-to-face**
- Provide a Q&A and supervised programming exercises on lecture topics
- One **one hour tutorial** every week
- Run by experienced teaching assistants
- **Bring your own device**
- In C60 Cyberphysical Lab
- Tutorials start next week (not this week)

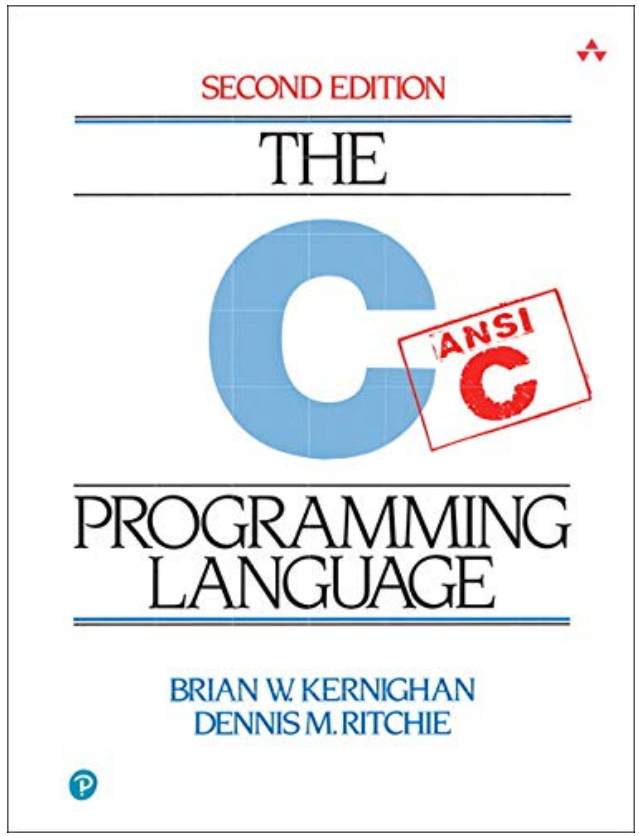
Assessment

- Coursework:
  - 75% of your overall module grade
  - practical programming problems (70%)
  - 6 programming courseworks in total
  - released and submitted on GitLab - **projects.cs.nott.ac.uk**
  - lecture package quizzes on Moodle (5%)
    - best 8/10 so no need to apply for EC if you miss one
- Exam (written):
  - 25% of your overall module grade
  - questions will be on any topic in the module
  - more details given towards end of module

```
quizzes_grade = (double) num_quizzes_passed / 8;
if(quizzes_grade > 1.0)
    quizzes_grade = 1.0;
quizzes_grade = floor(5 * quizzes_grade);
```

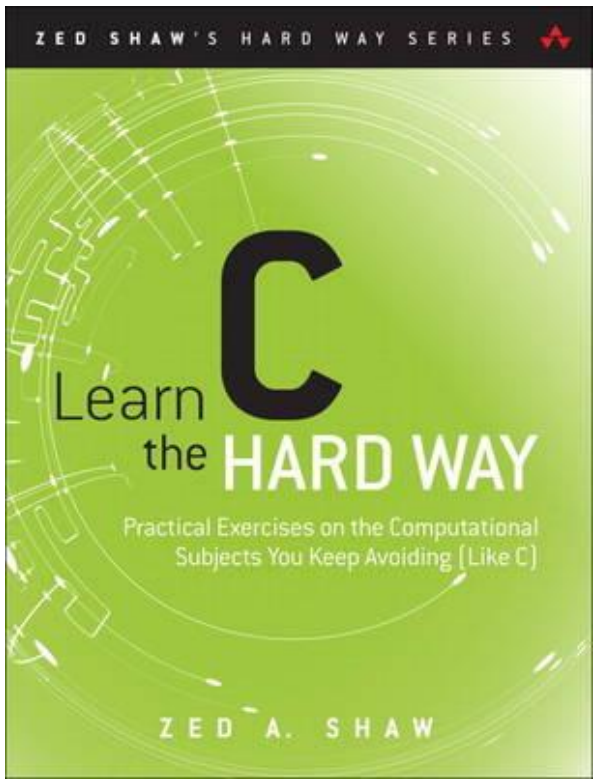
Programming Books

- The C Programming Language (2^{nd} edition)
  - Kernighan and Ritchie = **K&R**
  - ISBN: 978-0131103627
  - **essential** and worthwhile - buy it!



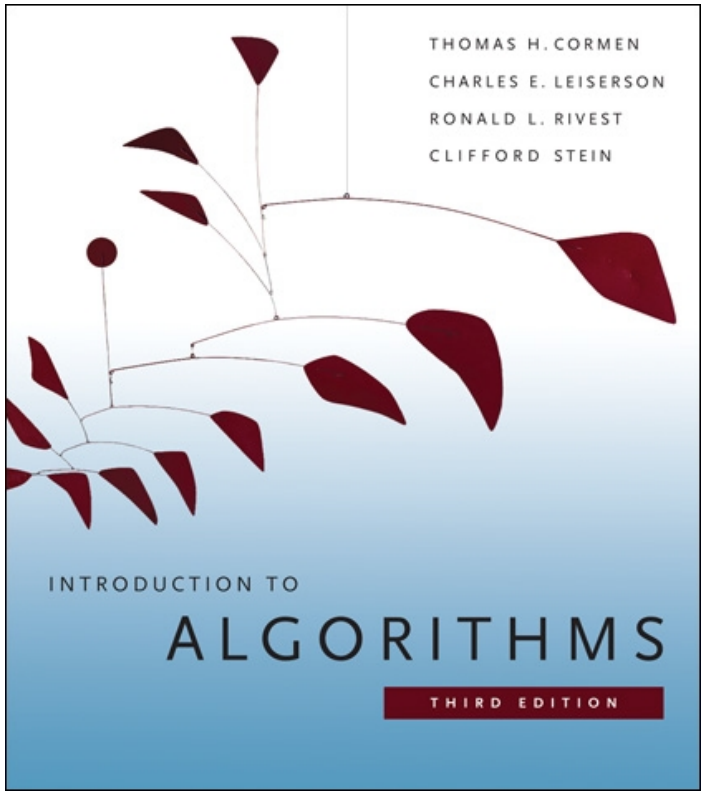
Programming Books

- Learn C the Hard Way: Practical Exercises on the Computational Subjects You Keep Avoiding (Like C)
  - Zed A. Shaw, Addison-Wesley
  - ISBN: 978-0321884923
  - **optional** - good for advanced programmers



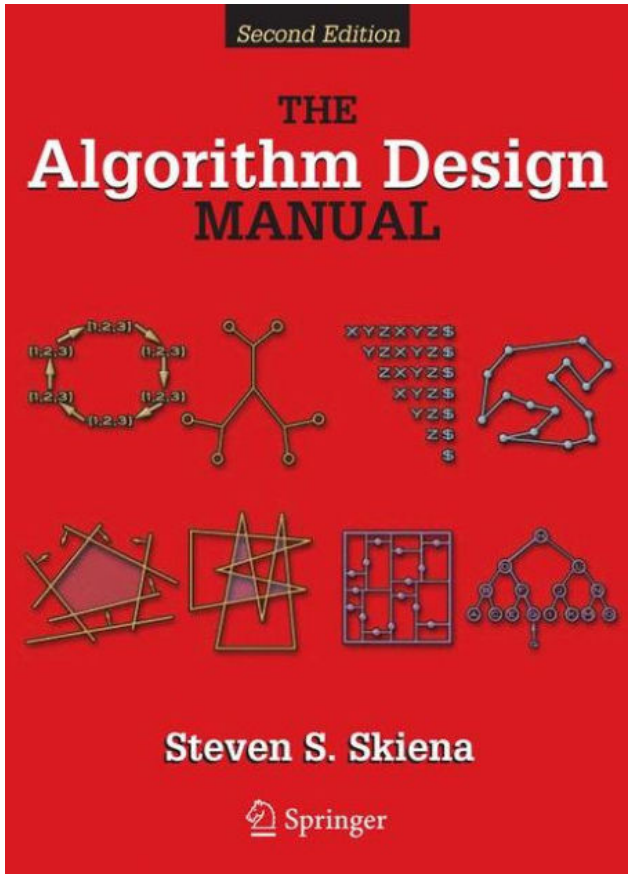
Algorithms Books

- Introduction to Algorithms (3^{rd} edition)
  - Cormen et al., MIT Press
  - ISBN: 978-0262533058
  - **recommended** - will be used in other modules too e.g. COMP2009 ACE



Algorithms Books

- The Algorithm Design Manual (2^{nd} edition)
  - Steven S. Skiena, Springer
  - ISBN: 978-1848000698
  - **alternative**



Software

- This module uses **Linux** and **open-source software**
- As discussed in Welcome to the Machine Session
- You can either:
  - **best**: run your own distribution
  - **OK**: use a VMware Horizon and the School's Linux Virtual Desktop
- Running your own Linux:
  - **Linux** users - you are there already ☺
  - **Mac** users - you are almost there - install **VirtualBox/UTM** or **Xcode**
  - **Windows** users - **VirtualBox** or **dual-boot**
- Any Linux distro will do: Debian, Ubuntu, Mint, SUSE, CentOS, Fedora ...

Summary

- Everything can be found on the **COMP1005 Moodle pages**
- **Lectures**: in person and pre-recorded
  - study on Monday 10:00-12:00
  - attend (B52) on Tuesday 10:00-11:00
- **Labs**: in person in A32 - start **this week**
- **Tutorials**: in person in C60 - start **next week**
- **Office hours**: Tuesday 11:00-12:00 in-person B48
- **Assessment**: 75% coursework, 25% exam
- **Course books**: K&R essential
- **Software**: own Linux or Linux Virtual Desktop

Activities

- Go though the COMP1005 Moodle page
- Get the K&R course book
- Install your own Linux or the VMware Horizon client
- Study this week's lecture package
- Do this week's assessed Moodle quiz