Course Introduction

- Staff
- Course Objectives
- Ways of Learning
- Lecture Schedule
- Assessment
- Avoiding Plagiarism
- Administration, lab locations
- Feedback from past sessions
- See the class home page
  www.cse.unsw.edu.au/~en1811

Course Objectives

- What you **should be able to do** by the end of the semester:
  - **use spreadsheets** and their associated tools to solve small computational problems in Engineering, Science and Business [topic SS];
  - design and implement solutions to problems by **writing small programs** using a scripting language such as OpenOffice Basic or equivalent [topic BP];
  - **use a numerical computing environment** such as MATLAB® to analyse, model and visualise data and systems [topic NC]; and
  - understand a bit about **where the technology is and might be going** (so you can be better prepared to exploit it in your professional career) [topic IT]

Ways of Learning

- Lectures
  - slides released as PDFs beforehand
  - only effective if audience cooperates by maintaining silence
  - will try to add occasional interactive elements (bonuses for volunteers!)
- Labs
  - develops experience with using the problem-solving systems
  - tasks based on previous lecture material
  - has a small on-line assessment exercise based on very basic knowledge
  - fully supervised and assessed within the class, some self-assessment
  - must be ready to show (some) work 30 minutes before end
- Assignments
  - completed in own time, individual unless specified
  - fully develops skills, especially in programming
  - important exercise in time management
  - submitted on-line, fully tested and objectively assessed
  - second assignment is peer-assessed using objective criteria
  - late penalty is 15% per day off the maximum available mark
- Revision lab classes
  - offered once or twice mid-semester, covering programming material
  - focus is on absolutely minimum level of knowledge to pass
- Consultation
  - lecturer has specific times, or see after class
  - lots more scheduled prior to assignment due dates
- Course forum
  - general, lecture, labs and assignment-specific
  - used for assignment-related questions and answers
  - tutors and lecturer will post and reply
  - strongly encouraged to participate
  - usual etiquette:
    - respect for participants’ opinions
    - please check before starting a topic that it’s genuinely new
    - no assignment solutions (tiny fragment is OK to ask a question though)

Lecture Schedule

- There is a one week break after week 5
- The two streams merge after week 7, all interested go to CLB-7

<table>
<thead>
<tr>
<th>Week</th>
<th>Topic</th>
<th>Lecturer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 to 3</td>
<td>SS: Spreadsheets and Data Analysis</td>
<td>Geoff, Tim</td>
</tr>
<tr>
<td>4 to 7</td>
<td>BP: Problem Solving and Programming (OpenOffice Basic)</td>
<td>Geoff, Tim</td>
</tr>
<tr>
<td>8 to 11</td>
<td>NC: Numerical Computing, modelling and visualisation using Matlab</td>
<td>Tim</td>
</tr>
<tr>
<td>11, 12</td>
<td>IT: Introduction to some of the current and emerging Information Technologies</td>
<td>Tim</td>
</tr>
</tbody>
</table>
Assessment

- Labs (weeks 2 to 13)
  - each lab has an on-line assessment (1 mark) and tutor’s assessment (marked out of 3)
  - best 10 marks for each category taken of potential 12
  - contributes 10% of total
- Assignments
  - assign 1 due w2 (OObasic program), 7% (but see below)
  - assign 2 due w11 (Matlab program), 8% (but see below)
- Mid-Semester Test
  - during week 5 lab, 45 mins, practical, 15%
  - covers first 3 weeks’ material (labs 2 to 4)
- Final written exam
  - 3 parts: multiple choice, OObasic, Matlab
  - 60% of overall assessment, minimum competency 33.3% (20/60)
  - Non-linear formula applied if min exam mark not reached (details later)

Curriculum/Assessment Matrix

<table>
<thead>
<tr>
<th>Topic</th>
<th>SS</th>
<th>BP</th>
<th>NC</th>
<th>IT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assign1 labs</td>
<td>3</td>
<td>3</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Assign2</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exam assign1</td>
<td>8</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exam other</td>
<td>10</td>
<td>10</td>
<td>15</td>
<td>7</td>
</tr>
<tr>
<td>TOTAL</td>
<td>28</td>
<td>30</td>
<td>35</td>
<td>7</td>
</tr>
</tbody>
</table>

There is one possible visualisation of the data (Matlab’s bar3 function).

Avoiding Plagiarism

- Academic honesty
  - everything submitted for assessment must be your own work
  - acknowledge all sources unless obvious
- Assignments 1 and 2
  - program code must be developed alone (unless explicit groupwork)
  - discussion about solutions OK, indeed encouraged
  - imperfect but honest attempt will still attract fair marks
- Anti-plagiarism measures
  - start early and get help if you’re struggling
  - we usually run sophisticated similarity analysis software
- Reference site (though essay-oriented)
  - www.lc.unsw.edu.au/plagiarism

Administration

- Changing classes
  - myUNSW used for all changes if possible
  - places are limited for this semester
  - intractable timetable clash is the only reason for squeezing into a full class
  - email the class account en1811@cse.unsw.edu.au
- Lab locations — Physics (Old Main) Building, OMB, behind Burrows Theatre (see map overleaf)

Lab locations

Feedback from past sessions [our emphasis]

- Best bits
  - “Very practical and interesting course.”
  - “The labs helped dramatically to cement the material covered in lectures.”
  - “Lots of practical tasks/problems with a real-world application/basis, with a strong focus on providing useful and practical knowledge.”
  - “It improves our thinking”, “online examples shown”
- Things that could be improved
  - “Explaining better to students who have never even used excel”
  - “more control by telling others to keep quiet in the lecture theatre”, “too much noise from other student[s]”
  - “more coverage of the basic aspects of vba programming. …most students are being exposed to vba for the first time.”
  - “providing more examples on how to function or create a program as I got really confused as to how to use some things.”
  - “more interactive lecture”, “making it more relevant”, “easy programming”, “none”
  - “assignment was too difficult”, “encourage more to do more pre lab work”
More on the Class Web page

- The class home page is the source of all official information about ENGG1811. Its contents include
  - Notices, updated frequently (keep a close eye on them).
  - The Course Outline (near the top of the left sidebar) which contains formal details about the course.
  - CourseWork: lecture notes and working documents, lab exercises and assignment requirements.
  - Help: Consultation schedule and course forum link.
  - Resources: online documentation, tutorial material and lecture recordings.
  - Timetable and lab class roll.
  - Assessment records and assignment submission/collection.
  - Staff and policies.
- The class web page (in case you missed it) is still at www.cse.unsw.edu.au/~en1811

Checklist

To start this course off on the right foot, make sure you have done all of the following by the end of week 1.

- Enrolled in the course properly (with a lab class)
- Found out where the labs are
- Installed OpenOffice 4 on your own computer (strongly recommended)
- Had a go at the first lab (lab 02) if you have OpenOffice Calc
  - Excel solutions might convert to OO Calc but no guarantees
  - The labs and midterm will use OO Calc so you should get used to it
  - Recycling solutions from previous semesters gets you no (or negative!) marks
- Dropped into the course forum, maybe posted a comment
- Considered buying the main reference book (recommended)
  - not used until week 8 so no hurry

Class home page (yet again): www.cse.unsw.edu.au/~en1811