Welcome!

COMP1511 18s1

Programming Fundamentals

— Lecture 0 —
Hello, World!

Andrew Bennett

<andrew.bennett@unsw.edu.au>

course introduction

the big picture

many sights to C

the bigger picture...

what is programming?

doesn’t have to be in a language like C…

https://www.youtube.com/watch?v=FN2RM-CHkuI

why is programming awesome?

What?

this is a course where you will…

learn to program

become a confident programmer

write code you're proud of

discover the joys of programming

It can be tricky at first

code won’t compile

not sure what’s going on

easy to lose track of what you’re trying to do

is “it works” good enough?

Who's teaching?

3x lecturers:

Mr Andrew Bennett

(that's me!)

Dr Andrew Taylor

Dr John 'jas' Shepherd

course convenor:

Dr Andrew Taylor

course administrator:

Mei Cheng Whale

tutors + lab assistants:

too many to list!

Who’s learning?

...

You!

Three types of students

red

yellow

green

Should I take COMP1511?

How?

Lectures

introduce theory and practice of programming

Tutorials and **Laboratories**

reinforce ideas and concepts with hands-on examples

Assignments

assess understanding of C, problem-solving skills

Weekly Coding Tests

regular and realistic feedback of your understanding of course content

Final Exam

a 3-hour theory and practical extravaganza, in CSE laboratories

Assessment

assignment 0

worth 6%, due week ~5

assignment 1

worth 12%, due week ~9

assignment 2

worth 12%, due week ~12

weekly coding tests

best 8 of 10, worth 8% total, weeks 3-12

lab exercises

groups, worth 12% total, due weekly

final exam

3h theory+practical exam, worth 50%, during the exam period

Communications

official communications from the course
will come to your UNSW email address
make sure you can receive emails!

if you set up email forwarding, **test it!**

when you send emails,
send them from your UNSW email address
and include your zID…
don't email from personal email addresses!

to get in touch with the course urgently
email `<cs1511@cse.unsw.edu.au>`

Course Forum

ask anything about the course / computing

receive answers from your tutors and classmates

link is on course website, or here: https://edstem.org/courses/1950/

Course Evaluation and Development

assessed with **myExperience** and the **Sturep Survey**

* * *

informal feedback during the semester is very welcome!

let us know of any problems as soon as they arise
we can't fix problems we don't know about

Conduct and Integrity

treat people with **courtesy** and **respect**
… we are **all** humans …

* * *

pretending someone else's work is yours is **not okay**.

CSE is a bit different to other places…
we don't care *how* you reference,
we just care *that* you reference

important: read the course outline!

More information?

course material lives on WebCMS3

webcms3.cse.unsw.edu.au/COMP1511/18s1

* * *

please read the **course outline!**

How to do well in this course

practice consistently across the **entire** course

prepare for all tutorials and labs
by attempting the questions **before** your class.

attend all tutorials and labs.
ask questions.
use your resources.

make a list of each compile error you get,
and how you fixed it.
(they will come back to haunt you repeatedly… this will be invaluable.)

Resources

optional textbook: Alistair Moffat

Programing, Problem Solving, and Abstraction with C

(Pearson Educational, 2003; ISBN 978 1 74103 080 3)

Google is your friend,

as is Stack Overflow,

especially when debugging compile errors

Getting Help

read the **course forum**

ask your questions there, if they're not answered

ask one of your COMP1511 peers

ask your tutor! (they are all very friendly :-)

talk to me after lectures

Getting Started

Before the end of this week, you should:

do **Lab 1**.

be able to write a ‘hello world’ program
from your **CSE account at uni**, and

be able to write the ‘hello world’ program
from your **home computer**.

The CSE Labs

CSE has lab computers…

unlike other workstations at UNSW,
these don't run Windows;
they run Linux, which is very different

the easiest way to use these
(if you're not in a lab)
is using VLAB

use your **zID** and **zPass** to log in
if you don't have a zID/zPass,
you should fix that asap!

It's All Text!

we write programs in a **text editor**
very different to (e.g.) Word or Pages

we'll be programming in **C**
which has well-defined rules for how the language works,
which means we can use this to describe
something that can be turned into a program
that the computer can run

let’s try it!

Hello World

```
// Prints out a friendly message.
// Andrew Bennett <andrew.bennett@unsw.edu.au>
// 2018-02-27

#include <stdio.h>

int main (void) {




    return 0;
}
```

Hello World

```
// Prints out a friendly message.
// Andrew Bennett <andrew.bennett@unsw.edu.au>
// 2018-02-27

#include <stdio.h>

int main (void) {

    // Print out the famous 'hello world' message.
    printf ("Hello, world!\n");

    return 0;
}
```

Navigating on Unix

pwd shows where you currently are

```
$ pwd
/import/ravel/2/andrewb
```

ls lists the items in the current directory

```
$ ls
18s1    bin     lib     public_html     tmp     web
```

mkdir makes a new directory

```
$ mkdir cs1511
$ ls
18s1    bin     cs1511  lib     public_html     tmp     web
```

Navigating on Unix

cd changes directory

```
$ cd cs1511
$ pwd
/import/ravel/2/andrewb/cs1511
$ ls
$
```

cd .. changes into the previous directory

```
$ cd ..
$ pwd
/import/ravel/2/andrewb
```

Writing a Program

to create a C program from the terminal,
open a text editor like **gedit**

```
$ gedit hello.c &
```

once the code is written and saved...
compile it with **dcc**!

```
$ dcc -o hello hello.c
```