CA170 Week 1:

Intro to Unix

The Command-line

- Text-based interface.
- Type commands with arguments at a "prompt":

```
        $ command (arg1) (arg2) ...
```

- You can use this as an adjunct to (rather than replacement of) the **GUI** interface.
- The "prompt" may be anything, not necessarily "$". It doesn't matter what the prompt is.
- Note: In notes I often use the convention "$" to indicate the prompt, no matter what the prompt is.
- Can launch GUI programs:

```
        $ gedit file &
```

PATH

- Most commands are not in-built, but are programs, found somewhere in the "PATH" variable:

```
        echo $PATH
```

- This makes the command-line "shell" a reasonably simple program:
- See if there exists in the directories listed in the variable $PATH an executable file with the same name as the 1st "argument".
- If so, execute this file, passing it the other arguments. Else print error message.
- Note: The complete list of executable files in the path is normally read once and cached in memory for fast access every time a command is typed.

Hierarchical Directory Structure

`pwd`	`Print working directory`
	`e.g. /users/staff/jsmith`

`cd`	`Change directory`
`ls`	`List files`

`cd ..`	`Go to parent directory`
	`e.g /users/staff`

`.`	`Current directory`
`/`	`root directory`
`$HOME`	`home directory`
`$HOME/public_html`	`public web space`
`/tmp`	`system temporary files`

- Hierarchical file system - /directory/sub-directory/file
- Forward slash (this is why Web is forward slash).