Labs 1-6

Some quick notes

Marking Schemas

- The Linux Labs should have approximate marks by each question.

- The C Labs don't have a full schema, as it's impossible to specify this without telling you how to do the questions.
  - For "programming" questions (where you write your own code):
    - 30% of the Marks are for Style, Layout and Comments.
      - See the note on these in the P2T moodle
    - 70% are for your actual source code. [Including if it compiles without errors or warnings]
    - Please submit C source files (.c files), not compiled code. Compiled code cannot be marked.
  - For debugging questions, usually 1 mark per issue.

Chained Questions

- In some of the later C Labs, there are exercises which are dependant on previous lab exercises from previous labs

- For these, we will make available a worked minimal answer to the previous question, for students who were unable to answer it.

- Students may use the minimal answer as a basis for the additional work needed in the dependant question.

- This won't happen until **Lab 4.**

Submitting Labs

- When submitting to Moodle, you should:
  - If you have just 1 file to submit, submit the file by itself.
  - If you have many files to submit, archive them into an archive file (zip, tar, etc)
  - Submit the archive.

  - To make a tar archive for a directory:

$ ls
Lab1 Lab2 Lab2

$ tar cvf Lab2.tar Lab2
Lab2/
Lab2/afile
Lab2/afile2

$ ls
Lab1 Lab2 Lab2.tar Lab3

Submitting Labs

- When submitting to Moodle, you should:
  - If you have just 1 file to submit, submit the file by itself.
  - If you have many files to submit, archive them into an archive file (zip, tar, etc)
  - Submit the archive.

  - To make a tar archive for a directory:

ls shows that we are in a directory containing 3 directories: Lab1, Lab2, and Lab3

```
$ ls
Lab1 Lab2 Lab2
```

```
$ tar cvf Lab2.tar Lab2
Lab2/
Lab2/afile
Lab2/afile2
```

tar takes three arguments:
a list of options, the name of the file to make, and the thing to archive
"cvf" are "**c**reate a tar file, **v**erbosely (say what you're doing), specifying a **f**ilename"
Lab2.tar is the name of the file we're making (needed because of **f** above)
Lab2 is the directory we're making the tar out of

Output of tar lists all the files it is adding to the tarfile. (In this case, Lab2 contains 2 files, afile and afile2)

```
$ ls
Lab1 Lab2 Lab2.tar Lab3
```