Systems Programming: Practical 1 — Programming in the Shell

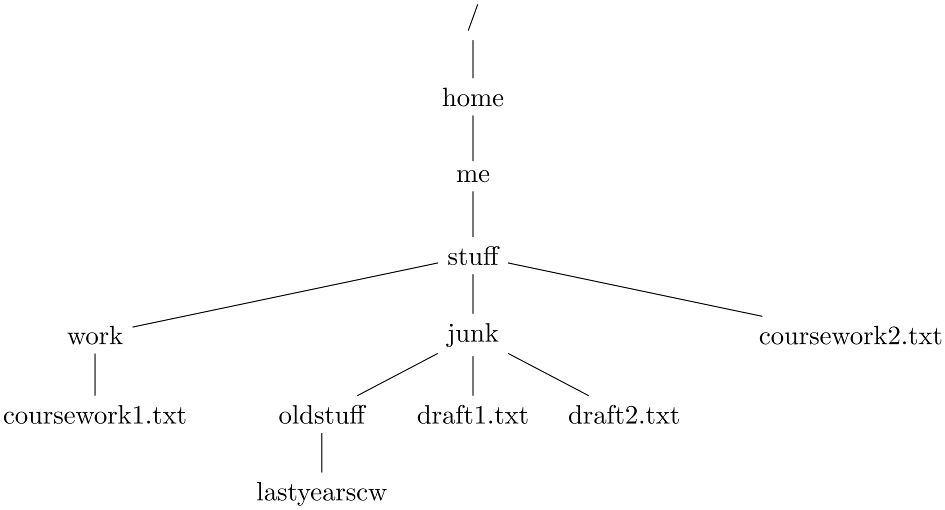
All these exercises need to be performed on a UNIX system or UNIX type system. You can use a MAC, a Linux distribution or remotely login to the university shared Linux system: `mira.dur.ac.uk` (see Practical 1 for instructions).

Github Classroom

Finish the introduction to git assignment on github classroom. This requires you to go to `https://classroom.github.com/a/oEye4ZgZ` and accept/complete the assignment. This assignment gives you a lot of additional resources on git if you need them.

A Directory Commands

Consider this Linux directory structure. You may assume that anything without the file extension `.txt` is a directory.



The user logs into the `/home/me/` directory and executes the following commands.

```
cd stuff
mv coursework2.txt work
cd junk
rm *
cd ..
mkdir admin
cd admin
touch todolist.txt
```

1. Draw the directory structure again, as it would be on completion of the above commands.
2. What would the output be if the user ran `pwd` immediately after the above commands?

B Dictionary

Computer dictionaries are often made by pulling each unique word from a document (or set of documents). In this exercise you will use a pipe of `tr` commands to extract the individual words from the novel "Pride and Prejudice" in order to generate a Jane Austin dictionary. The file `1342.txt.utf-8.txt` contains a copy of the novel. The steps in the pipe are as follows:

- Convert all newline characters to spaces
- Delete all non printable characters
- Delete all non-word characters: `0-9`, `_`, `"`, `;`, `{`, `}`, `[`, `]`, `(`, `)`, `-`, `!`, `:`, `#`, `$`, `%`, `@`, `^`, `&`
- Convert spaces to newline characters
- Convert all the letters to lower case
- Sort the words
- Keep only the unique words

C Shell Scripts

1. Write a shell script which, for all files in the current directory, prints out the string "The number of characters in the file" and the name of the file, followed by the number of characters in the file.
2. Write a shell script which takes in a single parameter and prints out "This number is bigger than 10" or "This number is not bigger than 10" depending on whether the parameter is bigger than 10 or not.
3. Write a shell script which takes a single parameter and makes a new directory called `MyTextFiles` if it does not already exist. If the parameter entered is `1`, the script should copy all text files (make sure you deal with all permutations of capitalisation, such as `.txt`, `.Txt`, `.TXT`) from the current directory to the `MyTextFiles` directory and if the parameter entered is not `1`, the script should move all text files from the `MyTextFiles` directory to the current directory. If the number parameter entered was not `1` and the `MyTextFiles` directory does not exist, you should give a warning to the user. In either case, the script should then produce a list of files within that directory piped to a new file called `DirectoryListing.txt` and display the number of lines in the file `DirectoryListing.txt` on the screen.

D Text translation

Write a set of `tr` commands to translate the file `file.txt` into the output shown below:

```
1,leaf,4.5,56,brown
2,needle,3.0,45,silver
3,desk,104.0,453,white
4,chair,56.5,124,magnolia
```

E Regular Expressions

Work through the regular expression exercises at: `http://regexone.com/lesson/0`