

# CS1110 Introduction to Systematic Programming

## 1st Practical Class Week 4

In this lab we will be looking at

### Use of Emacs Ada Mode File I/O in Ada

Copy the files

```
/usr/local/staffstore/cs1110/debug-exercises/mess.adb  
/usr/local/staffstore/cs1110/tutorial-programs/petrol.adb  
/usr/local/staffstore/cs1110/tutorial-programs/petrol.dat
```

to your Ada sub-directory<sup>1</sup> using commands of the form

```
cp /usr/local/staffstore/cs1110/debug-exercises/mess.adb ~/Ada
```

Remember to use Tab filename completion etc. to reduce the amount of typing involved in entering the long pathname. Also can you see how to use a wildcard to copy both 'petrol' files with one command?

Open the file `mess.adb` in Emacs. This is a badly formatted and buggy version of the example program from Unit 6. Use the commands

```
Adjust Case in File  
Indent Lines in File
```

from the **Ada Edit** sub-menu to format it correctly. The results in a great improvement in the layout. However as there are several typos in the file not all the lines will be indented correctly.

The file contains a few misspelt Ada keywords. Try to locate these and correct them by using the visual clues provided by capitalisation and syntax colouring

keywords should appear all in capital letters  
keywords should be coloured purple.

Starting from the top of the file look for strangely indented lines and then look for missing semi-colons, right brackets or closing quotation marks on the preceding lines. Correct each error by inserting the missing character. Then move the cursor down to the badly indented line below and press Tab to indent the line correctly. If the line indents properly you have successfully corrected the typo. If the line still does not indent properly, then try again to correct the typo! Continue until you have reached the end of the file.

If whole regions of the file are badly indented then use the command

```
Indent Lines in File
```

after correcting the typo.

Move the cursor to the start of the file and then replace `FLOAT` by `FLoat` everywhere in the file using **Query Replace** from the **Search** menu. Similarly replace `GET` by `Get` everywhere in the file.

When you think you have corrected all the errors, compile the program using **Build** from the **Ada** menu and run it. If necessary, correct any remaining errors and recompile.

PTO...

---

<sup>1</sup> If you don't have an Ada sub-directory create one as described in the hand-out Lab3.1

### Optional exercises

Try out some of the commands from the **Ada Goto** menu:

```
Goto Start of Statement
Goto End of Statement
```

Try out some of the commands from the **Ada Statements** menu:

```
While Loop          For Loop
If   Elsif          Else
```

### **File-IO in Ada**

The file `petrol.adb` is an interactive version of the program discussed in the tutorial last week. Edit this file so that it reads data from a file `petrol.dat` (say) and so that it sends its output to a file whose name is specified at run time.

You will need to

- remove the `Put` statements that output interactive prompts
- add appropriate `OpenInput` and `CloseInput` commands etc.
- add appropriate context clauses for the library `CS_File_IO`

Then compile and test the program.

Note a way to delete a whole line is to select it with the mouse and then press `C-w` (i.e. press the Control and w keys together). This is faster than using Cut from the Edit menu. Even quicker if the cursor is in the line concerned move the cursor to the start of the line with `C-a` and then delete the line with `C-k`. This will leave a blank line and then press `C-k` again to remove the blank line if required.