

CS1110 Introduction to Systematic Programming

First Practical Class

Do not attempt the work on this sheet until you have successfully completed the environment set-up steps as instructed in the induction week practical.

If you did not attend this practical or if you did not complete the worksheet, follow the environment set-up steps in the introductory hand-out BEFORE working on this sheet.

In this class we will be looking at

- basic use of the Common Desktop Environment (CDE)
- use of the Help Viewer to learn more about CDE
- basic use of the Emacs editor to create and modify files

Log on as discussed in the induction week labs. If you are not familiar with this process consult the introductory hand-out.

Exercise - Basic Use of the Window System

1. The **root menu** allows you to execute several useful programs - editors, web browsers, terminal windows and email tools. To access the root menu **left-click** on any part of the background - the background is that portion of the screen outside the windows and front-panel. Note that the menu contains several items which can be selected by moving the mouse pointer. By dragging the mouse pointer into the background the root menu will disappear - try it.
2. Use the root menu to open a new **shell** (or `xterm`) window. Bring up the root menu as in step 1 above - drag the mouse pointer down to the item **Shells** and then drag to the right when a pop-menu appears - select **xterm** and release the mouse button.

Note - it may take several seconds before the new window appears - **be patient** - the system is processing your command which may take some time to accomplish - note that whilst it is doing this the small globe on the front panel rotates to show the system is active. Wait for the globe to stop rotating before opening other windows.

3. Use the root menu to open some other items.

Exercises - Working with Windows

The above operations opened windows - these have a basic structure and means of operation which you must learn

1. Iconify the shell window that you have just opened by clicking on the minimize button - this is the left button in the top right hand corner of the window - this shrinks it to a small image (**icon**) - restore the original size of the window by double-clicking on the icon. Iconifying windows that are currently not in use reduces desktop clutter.
2. Maximise the size of the window by clicking on the maximize button - this is the right button in the top right hand corner of the window - return it to its original size by clicking again on the maximize button

Exercises - Moving and Resizing Windows

1. To move a window drag the title bar of the window to the desired position - the title bar is the top border of the window containing the title.
2. To resize a window drag the mouse from one of the corners - to resize a window in one direction (horizontally or vertically) drag the border

Note when windows overlap on the desktop a partially hidden window may be **raised** to the front by simply clicking on a visible portion of the window.

Exercises - Use of the Window Menu

Each window has its own menu - the **window menu**. To display the window menu click on the window menu button situated at the top left corner of the window - the following

items (among others) can then be selected by moving the mouse to the required item and clicking:

- Restore** - changes the icon back to a full window - can also double-click on the icon to restore the window - note that if the window is already open then the menu item will be greyed out meaning it is inactive and cannot be selected
- Move** - by moving the mouse the window moves accordingly - left-click to complete the move
- Size** - by moving the mouse the window is resized accordingly - left-click to complete the operation
- Minimize & Maximize** - same effect as described above
- Lower** - has the effect of moving the window behind any overlapping window
- Close** - has the effect of closing a window and removing it completely from the desktop - any text displayed in the window is lost unless previously saved. Note difference between Close and Minimize. A quicker way to close a window is to double-click on the window menu button.

The Workspace Menu

When the left mouse button was clicked in the background the root menu was displayed. If the **right** mouse button is clicked then the workspace menu is displayed - it contains 6 items:

| | | | |
|-------------------------------------|---------------------|----------------------------------|----------------|
| Shuffle Up | Shuffle Down | Refresh | Log out |
| Minimize/Restore Front Panel | | Restart Workspace Manager | |

Shuffle up and **Shuffle down** shuffle the windows up/down in the stack of overlapping windows - useful when there are several overlapping windows or completely hidden windows since repeated use of these operations brings windows to the front of the display one after another.

Refresh causes the whole workspace to be redrawn - useful when the workspace has become overwritten with system messages.

Minimize/Restore Front Panel is used to iconise and de-iconise the front panel.

Logout is an alternative to clicking Exit in the Front Panel.

Restart Workspace Manager causes the Workspace manager to be restarted - some changes made using the Style Manager (used to customise the behaviour of the CDE) do not take effect until the Workspace Manager is restarted.

Using Workspaces (WS)

The desktop is divided into four workspaces called One, Two, Three and Four. These can be used to run different applications thus allowing the desktop to be structured in a logical manner.

Normally workspace one is displayed on the workstation screen and the other WS are hidden. To display another WS, simply click on its button in the Front Panel - the required WS is displayed and new windows can be opened in this WS in the usual way.

Placing a Window in other Workspaces

To move a window to another workspace, click on the window menu and **select Occupy Workspace** - in the dialogue box that appears select the required workspace number - finally click on the **OK** button to complete the operation.

Note - if you change your mind then simply click on the selected item again when it will be deselected -.

To display a window in more than one WS hold down the Control key when selecting a WS with the mouse. If you want a window displayed in all four WS then **select Occupy All Workspaces** from the window menu.

Exercise - Desktop Help Viewer

The help Viewer provides on-line documentation for CDE. When you logon to the system a copy of the Help Viewer is displayed. Browse through the documentation to learn more about CDE and its advanced features. For example you can change the background of the desktop, change fonts used to display information in windows etc.

Exercise - a brief introduction to the Emacs Editor

1. We are going to use Emacs to create a new file called `MyInfo.txt` (say) and use this file to store some information about yourself.

To start the Emacs editor start Emacs from the **Editors** section of the **root** menu (use the menu item "**Emacs plus filename prompt**") and type the name of the file `MyInfo.txt` (say) when prompted to do so

OR type into an active `xterm` window:

```
emacs MyInfo.txt &
```

2. Make sure that the Emacs widow has the focus - type in some information into the Emacs window, say your name, address, course information etc and then save the result in a file called `MyInfo.txt` (say) by selecting **Save Buffer** from the Emacs **Files** menu.
3. This information is then stored in the file `MyInfo.txt` and can be retrieved at a future date - the file is stored in a space reserved for your personal files and called your **home directory**.
4. Quit the Emacs editor by selecting **Exit Emacs** from the **Files** menu or by selecting **Close** from the window menu.
5. To edit the file you have just created, start the Emacs editor again as in step 1.
6. Make some changes to the file e.g. type in your birthdate and then save the updated file under a different name by selecting **Save Buffer as** from the **Files** menu. - and type the new file name (`MyInfo2.txt` , say) in the mini-buffer at the bottom of the Emacs window.
7. Exit Emacs by choosing **Exit Emacs** from the **Files** menu.

We will now use Emacs to create a text-file called `prog1.adb` containing a simple Ada program.

There are in fact other ways of starting Emacs:

- 1) select "Emacs" from Editors submenu of the CS Root menu. When Emacs has started, select **Open File...** from Emacs' **File** menu and then type the name of the new file `prog1.adb` when prompted in the mini-buffer.
- 2) OR in an `xterm` window type

```
emacs &
```

and press return. When Emacs has started, select **Open File...** from Emacs' **File** menu and then type the name of the new file `prog1.adb` when prompted in the mini-buffer.
- 3) OR if you have Emacs already running, there is no need to start another copy -- just select **Open File...** from Emacs' **File** menu and then type the name of the new file `prog1.adb` when prompted in the mini-buffer.

Now type CAREFULLY the program below (which appeared in Unit 1) into the Emacs buffer:

```
WITH CS_Int_IO;    USE CS_Int_IO;

PROCEDURE Add IS
```

```

    First  : Integer;
    Second : Integer;
    Sum    : Integer;
BEGIN
    Get(Item => First);
    Get(Item => Second);
    Sum := First + Second;
    Put(Item =>Sum);
END Add;

```

As you type, the characters that you enter are stored in the Emacs edit buffer (temporary storage within Emacs) as well as being echoed on the screen. Note as you type Ada reserved words such as WITH, BEGIN etc. are displayed in a different colour and if you type them in lower case Emacs will convert them automatically to upper-case -- Emacs has gone into in **Ada-mode** and has many features for helping to format Ada programs. Emacs goes into Ada-mode because we specified the extension '.adb' (short for **ada body**) to the filename when we opened the file to indicate the file contains an Ada program. More on Ada mode in later hand-outs

When you have finished typing, save the program by selecting '**Save Buffer**' from the Emacs file menu. This will save the contents of the edit buffer in a file called prog1.adb on the computer's central magnetic disk so that it is available for future use. **You will need this program in the next lab class.**

Editing an existing file with Emacs

Again there are several ways of doing this:

- 1) If you already have an Emacs process running, select **Open File...** from Emacs' **File** menu and then type the name of the required file (prog1.adb, say) in the mini-buffer¹.
- 2) If no existing Emacs is running, start Emacs in any of the ways described above specifying the name of the existing file.

The buffer can now be edited in the standard way.

To set the insertion point (at a point other than the start of the file), left click at the point in the Emacs buffer at which you wish to insert extra text. This sets insertion point and then type in the required text. You may also use the 4 cursor 'arrow' keys to move the insertion point; this often more convenient than using the mouse for small adjustments of the insertion point.

You can use the delete key to delete the character immediately before the insertion point. This is useful for deleting small portions of text. To delete larger portions of text: left drag the mouse over the text that you want to delete and then select **Cut** from the Emacs' Edit menu

To move a block of text from one place in the edit buffer to another try using **Cut** and **Paste** from the Emacs' **Edit** menu:

left drag the mouse over the text that you want to select
 choose **Cut** from the **Edit** menu -- this removes the selected text from the buffer
 move the insertion point by left clicking
 choose **Paste** from the Edit menu to paste in the selected text at the new location.

You may copy text from one place in a buffer to another by repeating the above process, but choosing **Copy** instead of Cut at step 2. An alternative way (slightly faster) of copying text is

left drag the mouse over the text that you want to select
 move the mouse to the location where you wish to insert the copied text and **middle-click**. (Note that with a two-button mouse you 'middle-click' by clicking with both buttons simultaneously)

¹ If you want to edit two (or more) files at the same time, avoid starting several distinct Emacs processes (from the CS Root menu or from an xterm window); instead use **Open File...** from Emacs' **File** menu to open the second file in a new buffer and use the **Buffer** menu to switch between buffers. This saves computer resources and time.

Experiment with these techniques! Don't be afraid about making a mess. At any stage (until you save the changes to disk by selecting **Save Buffer** from the File menu) you can revert to the original version of the file by selecting **Revert Buffer** from the Emacs **Files** menu

Making prog1 more 'user friendly'

Using emacs, add suitable inforamory `Put` statements as described in Unit 2 at appropriate points in `prog1.adb`, for example insert:

```
Put(Item => "Please input the first number ");
```

immediately before the first `Get` statement and

```
Put(Item => "Please input the second number ");
```

immediately before the second `Get` statement and finally add the line

```
Put(Item => "Their sum is ");
```

immediately before the statement `Put(Item =>Sum);`

You will also need to add the following line right at the start of the program:

```
WITH Ada.Text_IO; USE Ada.Text_IO;
```

in order to make the library `Ada.Text_IO` available.

Now save this file as `prog2.adb` by selecting '**Save Buffer As...**' from the Files menu and typing in the new name `prog2.adb`² in the message-buffer. Note that the name of the buffer changes to `prog2.adb` and any future "Save Buffer" commands will save the buffer in the file `prog2.adb`.

Quitting

To quit Emacs, select **Exit Emacs** from Emacs' **Files** Menu.

Emacs has a useful safety feature: suppose you modify a file and try to quit Emacs before saving the changes to disk, Emacs will ask you whether you want to save the file before quitting. You can answer yes or no as appropriate. Thus you can't inadvertently lose a lot of work by accidentally quitting Emacs without saving your work.

Important Reminder -- Logout

Always logout at the END of every practical session so that other users cannot access your computer account and get up to mischief with the potential for doing untold damage. Remember you logout by clicking Exit in the Front Panel.

Note that the system remembers the 'state' of your desktop so that next time you access the system the desktop is displayed in exactly the same state left. So close down any unnecessary windows before exiting.

If you don't have time to complete this sheet in the lab class, please try to complete it in your own time.

In yur own time also try using the Help Viewer to find out about the Style Manager and other features of CDE.

² If you save the buffer by selecting 'Save Buffer' instead of 'Save Buffer As ...' then you will overwrite the old version of `prog1.adb` with the newer version -- not a major tragedy in this case!