Practical 0:
An Introduction to DICE and to the C compiler

Instructions

This Practical is assessed. In the course of working through the practical, you will create and modify some files. Credit will be given for submitting these for marking as described in Stage G. The deadline for completing and submitting the practical is **2pm, Monday, 4th October (week 3)**.

Aims

This practical is intended to introduce you to the computing environment (a `Linux`-based environment called DICE) provided for this course. You will need to understand everything in this practical in order to complete any later practicals.

Assessment

We will use your submitted files to determine whether you have completed the tasks satisfactorily. The maximum credit that can be obtained is 10 marks (from a total of 100 for the whole of the continuously assessed element of the course).

Prerequisites

You should have read the Computing Regulations. These are available, together with lots of other "New User Information", from the New User webpage[1] of the School of Informatics.

Notes

1. You need to understand all the material in this practical to complete the rest of the course. The course lecturers (and, when tutorials start, your tutors) will help you if you ask them. If you don't understand something please ask somebody: that's what they're there for.
2. You should use the workstations in the labs on Levels 5 of Appleton Tower (AT) for this practical. Please do not use lab rooms when there is a scheduled tutorial for another course.
3. This practical is divided into stages. In total, it should not take more than a few hours. Though short, it is very important - if you have problems, **get help early**!
4. Don't be afraid to try experimenting to see if you can work out what is happening — you shouldn't be able to damage the system by typing commands.[2]

[1] http://www.inf.ed.ac.uk/systems/newusers.html

[2] However, your attention is drawn to Section 2 of the Computing Regulations.

Stage A Logging in and out

How to start and end a session on the DICE machines in the labs on level 5 of AT.

Logging in

Find yourself a free terminal.

The login window presents you with `Username:` prompt. Your user name is the letter `s` followed by your matriculation number. Type your username at the prompt and press Return (or click on "OK").[3] Almost immediately, the `Password:` prompt will appear, indicating that you should supply the appropriate password. (Your password can be obtained from the Informatics Teaching Office (ITO) on level 5 of Appleton Tower, or alternatively in our scheduled Lab on Monday 27 September.) Type your password and again press Return .

After a short wait the system will display a collection of small pictures (called *icons*) and boxes (called *windows*.) The windows you see will depend on what programs were running when you logged out previously. This will always include the following:

- the **Panel** along the top of the screen with menus for Applications, Places and System, and with individual icons for the Firefox Browser, the Evolution email reader, and the Terminal console;
- a bar along the bottom of the screen (showing tabs for any open applications).

This environment (which arranges the windows and toolbars on the Desktop) is a Unix-based environment called Gnome™.

Logging out

The **Logout** option can be found in the `System` menu on the top toolbar. When you use this item, a *dialog box* will pop up, giving you the option to cancel the log-out if you want.

What you should now know

- ☐ How to log in to the Informatics system;
- ☐ how to log out from the Informatics system.

Make sure that you know how to do the above tasks (you can tick the boxes when you are sure that you can); you will not be able to complete the rest of the course otherwise. Ask someone if you have any problems.

The most likely problem you would have in these early weeks is that your password might not work. That may be due to accidentally having the `Caps Lock` or `Num Lock` switched on, or mistaking the letter 'O' for the digit '0', or something similar. In rarer cases, it may be that there is a problem with your password. In the latter case, contact the ITO.

Always log-out before leaving the lab.

[3] The "Return" key is marked "Enter" on some keyboards, or with a bent arrow on others.

Stage B Web Resources

In this stage you will check out and bookmark some key web resources which will be helpful for CP1.

Preparation

Read and make sure that you understand the Computing Regulations before accessing the web. These can be found as a link from the following "New Users" webpage of the School of Informatics:

`http://www.inf.ed.ac.uk/systems/newusers.html`[4]

To check out this webpage, note that you can start a browser by clicking on the Web Browser icon in the **Panel** at the top of your screen. (You should accept conditions of use if asked.)

Now look for the "New Users" webpage. Click on the **File** button at the top left of the browser, and the Select **Open Web Location** from the menu that appears. Then enter the following address (URL)

`http://www.inf.ed.ac.uk/system/newusers.html`

Then use the link for Computing Regulations and read these regulations. In particular, note that:

- Except for completing this section of this practical, surfing the internet is a low-priority activity. Anyone requiring a terminal for scheduled practical work has priority.
- Certain standards of decency apply — for example, accessing abusive, racist, pornographic or sexist material is prohibited.

Students are allowed access to the Internet in the expectation that they will behave responsibly. If these facilities are abused, your computing privileges will be restricted or removed.

Bookmarks are used to keep addresses that you want to be able to retrieve quickly. Move back to the "New Users" webpage, then lick on the **Bookmarks** icon, and then on **Bookmark This Page** to record the address of the 'New Users" webpage in your list of bookmarks.

There are a collection of other webpages which will be useful at various times during CP1. You should take a look at each of the following pages, and bookmark every one of them:

- The CP1 teaching webpage will contain all lecture notes, tutorial sheets and practicals as they are released. It also often has important announcements.

  `http://www.inf.ed.ac.uk/teaching/courses/cp1`[5]

- The Informatics Teaching Organisation (ITO), which has offices on Level 5 of Appleton tower, performs the administration of all Informatics courses. Their webpage is at the following address:

  `http://www.inf.ed.ac.uk/admin/ITO/`[6]

  The two links on that page that are most likely to interest you are the `ITO contact form` (for contacting the ITO) and the details of the `Informatics Computing Labs`.

[4] http://www.inf.ed.ac.uk/systems/newusers.html
[5] http://www.inf.ed.ac.uk/teaching/courses/cp1
[6] http://www.inf.ed.ac.uk/admin/ITO/

- The Informatics Support team, who do not have formal offices, are a group of people who maintain and support the DICE network. Their webpage is at the following address:

  `http://www.inf.ed.ac.uk/systems/`[7]

  The link on that page which will be of most importance is the `User Support FAQ`, which contains answers to almost every question you can think of. In the case of faulty equipment, you may use the `Support Request Form` to contact support.
  Remember that support do not help with any CP1-related questions, that is not their role.

What you should now know

☐ The details of the Computing Regulations for Edinburgh;

☐ Where the various key webpages important for CP1 can be found.

Make sure that you have all the aforementioned webpages bookmarked: they may be very useful in time to come.

Stage C The Window System

In this stage we will discuss the use of windows on DICE, with special reference to Terminal windows.

The desktop

The CP1 system uses a window system based on the *desktop* metaphor. Along the top of the screen is the **Panel**, which allows convenient access to many utilities. Between the top toolbar and bottom toolbar is the part of the screen referred to as the "desktop".

The **Panel** across the top of the screen contains three "icons" appearing after the three menus. The rightmost icon represents the "Terminal Command line". Click on this icon to launch a terminal emulator, also known as a *shell window* or a *console*.

Windows

As you become used to working on DICE, you may end up having a few windows open on the Desktop at any given time. As you probably know, to select a particular window to work with you just need to move the cursor into that window, and click on the window. The system confirms your choice by filling in the title bar of the current window in blue and greying the title bar of any other window.

[7] http://www.inf.ed.ac.uk/systems/

Unix commands

If you have been following these instructions, you have just now created a Terminal window. Make the shell window that you just opened into the current window and try typing some characters from the keyboard. You can remove them using Backspace . [8] Now try out some real (Unix) commands in the Terminal window:

(i) Type `ls`, followed by Return , to see which files are in the current directory (by default, this will be your personal *home directory*). This will probably show nothing (since you have not created any files yet).

(ii) Type `mkdir prac0` to create a directory for this initial practical. To make sure this has succeeded, use `ls`, followed by Return , again (to get extra information, use `ls -l`).

(iii) Note that you are *still* located in your original home directory. Type `cd prac0`, followed by Return , to move into the newly-created directory. From this location, you can add new files or sub-directories.

(iv) To ever return to your personal home directory, just type `cd ~`, followed by Return , at the prompt.

Apart from these basic commands mentioned above, please test out some of the other commands on the Unix sheet handed out at lecture 3.

Manipulating windows

The title bar of a window allows you to manipulate it in various ways. You can control a window's position, size, representation, and how it overlaps with other windows just by using the mouse and the title bar.

Windows have buttons on their title bars. On the left side, there are various menu buttons. On the right side, there is an "minimize" (or "iconify") button, a "maximise" button and a "close" button.

Try the following:

Iconify: You can close or iconify a window using the "minimize" button (looks like a "-") on the titlebar. This will make the window disappear, but it still retains a presence (in a lighter shade than before) as a button in the bar across the bottom of the screen. To iconify a window, position the mouse in the "-" near the right-hand end of the title bar and use the left or middle mouse button. The window appears to "disappear into" its button on the **Panel**. Try this with the shell window. To get the shell window back, simply click on its button on the **Panel**.

Closing: The "close" button ("X" marks the spot) on the title bar allows you to kill the window and the application that is running in that window. A dangerous option! Try deleting the shell window and restarting it from the icon.

Moving: Point the mouse near the centre of the titlebar and press and hold down the left mouse button. While continuing to hold the mouse button down, move the mouse around; notice that the window moves with the mouse. You can move any window this way.

[8] The "backspace" key may be marked with a *long* left-pointing arrow.

Resizing: You can use the outer frame of a window to resize it. Move the pointer so that it lies over any corner of a window (at which point the cursor becomes a double-headed arrow) and hold down the left mouse button. Now move the mouse while still keeping the button pressed. The size of the window changes as you move the mouse. When you release the button the window stays at this new size. Try it on one of your windows.

Maximize: Sometimes you want the window to be as big as possible. This can be quickly and easily achieved by clicking on the maximise button (the one marked with a square). The window expands to fill as much of the screen as possible. Clicking on the maximise button again returns the window to its original size.

What you should now know

- ☐ how to use simple Unix commands for files/directories in a Terminal window;
- ☐ how to open and close windows;
- ☐ how to move and resize windows.

Make sure that you know how to do the above tasks.

Stage D Changing your password

In this stage you will

- set a secure password for your account.

The password you have been given is unlikely to be easy to remember, and you should change your password to one that is. The requirement of being memorable (to you) is somewhat in tension with the other main requirement, namely that it should be hard for others to guess. Names of people and places are notoriously easy to guess (particularly with computer assistance). Passwords containing a mixture of digits, punctuation symbols, and upper and lower case letters are recommended. Finally, your password should contain at least six characters.

To change your password, run the program `passwd` in the shell window (remember: "Terminal Command Line" on the **Panel** to get a shell window) and reply to the prompts. You can change your password as often as you wish. Don't write it down! (The password is not stored in the computer. Instead, an encrypted form is stored. When you login, the password you type is encrypted, and the result is compared with the stored version.)

What you should now know

- ☐ How to change your password in the shell window.

Make sure that you know how to do the above task (don't tick the box until you are sure that you can).

Stage E Using the editor

In this stage you will find out

- how to use the editor to view and modify files;
- how to print files;
- how to create files.

Saving a given file

It is very likely that this task has been completed already! To accompany this Practical, there is a file called `questionnaire` which is to be downloaded from the course webpage. Two steps to be carried out:

- Go to the course webpage and use the right button of your mouse to click on the link for `questionnaire`. Take the `Save as ...` option, and save it into your *home directory* (starting `s`...);
- Use the `mv` command to move this file from your home directory into your `prac0` directory, before working on it. This command must be executed in the terminal window, from your home directory - then assuming you *already* have previously created the subdirectory called `prac0` (done during Stage C), just type `mv questionnaire prac0`, followed by Return. You can use the `cd`, `ls`, and `ls -l` commands to make sure that `questionnaire` has ended up inside the directory `prac0`.

Starting the editor

To start the editor, make sure you that you are in the directory which holds the file `questionnaire`. Then type `emacs`, followed by Return, in the shell window, and the system will load-up an `emacs` window.

Alternatively, use the bar at the top of the desktop and click on the sequence `Applications` → `Accessories` → `Emacs text editor`. This will load up the exact same editor.

Using the editor

The creation and editing of files is an activity which occupies a sizeable fraction of a computer user's time. This practical gives you an opportunity to learn your way around the editor (which is a variety of a standard editor called *emacs*) provided for you. It is worth taking some time over this: proficiency in the use of the editor will pay dividends later on in the course.

In this part of the practical you will use the mouse and menus to issue commands to the editor. The editor menus appear just underneath the title bar at the top of the `emacs` window. The menu entries often have keyboard equivalents (key combinations or key sequences that achieve the same effect). If a keyboard equivalent exists for a menu entry, it is given in the right-hand column of the individual menus.

Opening files

To open a file, pull down the `File` menu near the top-left corner of the editor window, slide the pointer to `Open file` and release the mouse button. You can now type the name of the file to be opened; as you type, its name will appear along the bottom line of the emacs window. Note - the starting directory for loading `emacs` was important:

- If you opened `emacs` from the Terminal window while inside the directory `prac0`, then you only need to type the extra letters `questionnaire` here;
- If you loaded `emacs` via the specified sequence from the `Applications` menu, the default directory will be your home directory. Hence you will need to type `prac0/questionnaire`.

When a file is opened, it becomes a *buffer* in `emacs`. You can open several files at a time (they become several buffers) and use the Buffers menu to switch between them. If you find that you have too many buffers, you can delete buffers (i.e., "close" a file) by selecting Close Buffer from the File menu. As we will see later, not all buffers correspond to files.

Moving and searching in files

Your editor window should now be displaying the file `questionnaire`.

Try out the four arrow keys (or *cursor* keys) at the bottom right of the keyboard, and watch how the cursor moves in response. What happens if you press the $\rightarrow$ key when you are already at the end of a line? Try moving forwards and backwards through the document with the $\uparrow$ and $\downarrow$ keys.. What happens if you press the $\downarrow$ key when you are at the bottom of the window?

Move to the beginning of the file using the keys mentioned above and use Ctrl s to find the first occurrence of the word "practical". Press Ctrl s first, then type `course` in response to the

I-search:

prompt. Do not press Return after typing `practical`.

Now press Ctrl s again and keep pressing Ctrl s until the command fails. Now stop the search process by pressing Ctrl g twice. Notice that you return to the beginning of the file (actually you return to the point from which the search was started).

Go to the beginning of the file again, and experiment to find out if the Ctrl s command is affected by the *case* of the string you tell it to find. In other words, would it have made a difference if you had typed

I-search: `praCTical`

instead of

I-search: `practical` ?

Stop the search this time by pressing Return . Notice that in this case you stay at the position of the last match found.

Modifying a file

Make sure you have the editor window open, and resize it so that you can see as much of the file as possible. (Hint: the maximise button can be used here.) You are now going to edit `questionnaire` with your responses to the questions there. You do this by typing from the keyboard, and using the keystrokes your have already learned to help you move up, down and around the page.

Take a look at the Edit menu. The Cut and Paste commands you learned above are duplicated there, as well as two related commands:
Copy

- lets you duplicate the selected region by making it available for pasting without cutting it.

Clear

- lets you remove the selected region by cutting it without making it available for pasting.

Notice how the menu items are originally greyed out: these commands are available unless you have already highlighted some region.

Test these commands by putting the answer to one question in the wrong position initially, and then use these commands to move your answer below the appropriate question.

There's also a very useful Undo command in the Edit menu that will undo your last editor command (be careful with this one - but it can be very useful if you Clear something by mistake).

Saving a file

Choose Save from the File menu to save the file `questionnaire` when you have finished editing it. You will only be entirely finished when you have also completed Stage F. However, you can answer some of the questions, save your file, and come back to edit it again later.

Setting up a default printer, and then printing your file

Often you will want to print out the files you are edited in `emacs`. However *before* you do this, you have to explicitly set-up a *default printer* - otherwise you will never see your printout. There are instructions about how to set up a default printer in the FAQ from the Informatics Support webpages. Here are the details.

- For your default printer, you will want to use one of the printers in Appleton Tower Level 5. there seem to be two printers on this level, named `at3` and `at13c`. Please *walk around* the open area of Level 5 to check I am right about this, then choose one of those to become your default printer.
- Then go to your Terminal window, and type *one* of the following at the command line (followed by Return , of course):

```
echo "export PRINTER=at3" >> ~/.benv
echo "export PRINTER=at13c" >> ~/.benv
```

This sets your default printer. You will have to *log out, and then log back in again*, in order for this info to be available to `emacs`.

After logging-out, and logging-in again, then re-open `emacs` to show `questionnaire` and choose Print from the File menu to print your file.[9]

Filling in the questionnaire

When you have finished Stage F, come back to finish editing `questionnaire`, and then electronically submit the `questionnaire` as explained in Stage G at the very end of this document.

What you should now know

- ☐ How to open a file for editing
- ☐ How to move about and search in a file
- ☐ How to insert text into a file
- ☐ How to delete text from a file
- ☐ How to cut and paste regions of a file
- ☐ How to save a file
- ☐ How to set up your default printer
- ☐ How to print a file from `emacs`

Make sure that you know how to do the above tasks (you can tick the boxes when you are sure that you can): you will have difficulty with the rest of the course otherwise.

[9] The printout will appear on your default printer.

Stage F Poem program

This is the fun part!

In this final stage of Practical 0, the goal is for you to write a small variation on the "Hello World" program which we discussed in Lecture 2 of this course. Your mission is as follows:

> You must write a short 4 or 5 line poem, somewhere containing your own name. Then you must create a small C-program named `mypoem.c` to print out the poem line-by-line.

Some initial comments:

- It will be very helpful for you to have the lecture notes from lecture 2 (Tuesday 21st Sept) at your side while you are doing this Stage of Practical 0.
- The poem does not have to be a masterpiece, just in your own words (and keep it appropriate).
- The program must print the poem out on a line-by-line basis.
- The program will not need to have any variables - it shares that feature with `hello.c`.
- You must check that your program compiles (using the `gcc` compiler), and also you must run the program to check the output is correct.

Step by step, here are your instructions.

(i) Create your poem before you make an attempt to write the C-program.

(ii) Open a new file `mypoem.c` in the `emacs` editor. To do this you just need to open `emacs` as in Stage E, and then type a couple of stray characters, then do an initial saving (when saving, make sure to save it as `mypoem.c` within the `prac0` directory; ie, you should see `/prac0/mypoem.c` at the bottom of the `emacs` window before pressing Return to save).

(iii) The first stage of writing any C-program is getting the basic structure (header files, `main` etc) of the program correct. Therefore as your first step, consider the simplest ANSI C program imaginable, and edit `mypoem.c` to contain this program:

```
#include <stdlib.h>

int main(void)
{
   return EXIT_SUCCESS;
}
```

Figure 1: The simplest C program

(iv) The header files of this very simple program (which does no processing and prints no output) are very limited. We will be using `printf`, so we will need to also include the `<stdio.h>` header file. Add the extra line for this and save again.

(v) Design a series of `printf` commands which will print out your poem line-by-line, and add these inside the `main` function in the appropriate order. Once you think you have a correct program, save again and exit `emacs`.

(vi) Type `gcc -Wall mypoem.c`, followed by Return , while you are in the directory which contains `mypoem.c`. This may give some warnings and/or errors to be corrected. If so, be proud that you have reached the DEBUGGING stage.

(vii) Once you have debugged your code and created an `a.out` file in your directory, type `./a.out`, followed by Return , to *run* your program!!! When you see the output, you will know whether your program needs extra modification.

(viii) When you are happy with your program, congratulations! Now you only need to finish filling `questionnaire` and then submit two files as described in Stage G.

What you should now know

- ☐ How to use the `printf` command to print formatted lines of text
- ☐ How to to use `gcc` to compile a C-program
- ☐ (maybe) How to interpret errors/warnings from the `gcc` output to help debug a program
- ☐ How to run the output of `gcc`

Stage G What/how to submit

So that we have some evidence that you completed this practical successfully, you should submit the files `questionnaire` and `mypoem.c` electronically. Assuming you are in the directory `prac0` (and assuming that this is the directory containing those two files!), you can do this by issuing the commands

```
submit cs1 cp1 P0 questionnaire
```

```
submit cs1 cp1 P0 mypoem.c
```

in the shell window.[10]

[10] `cs1` indicates first year, `cp1` indicates the course, `P0` the practical, and `mypoem.c` (or `questionnaire` respectively) the file to be submitted.