Lab1: UNIX Basic

EC327

"Introduction to Software Engineering"

Fall 2015

OVERVIEW

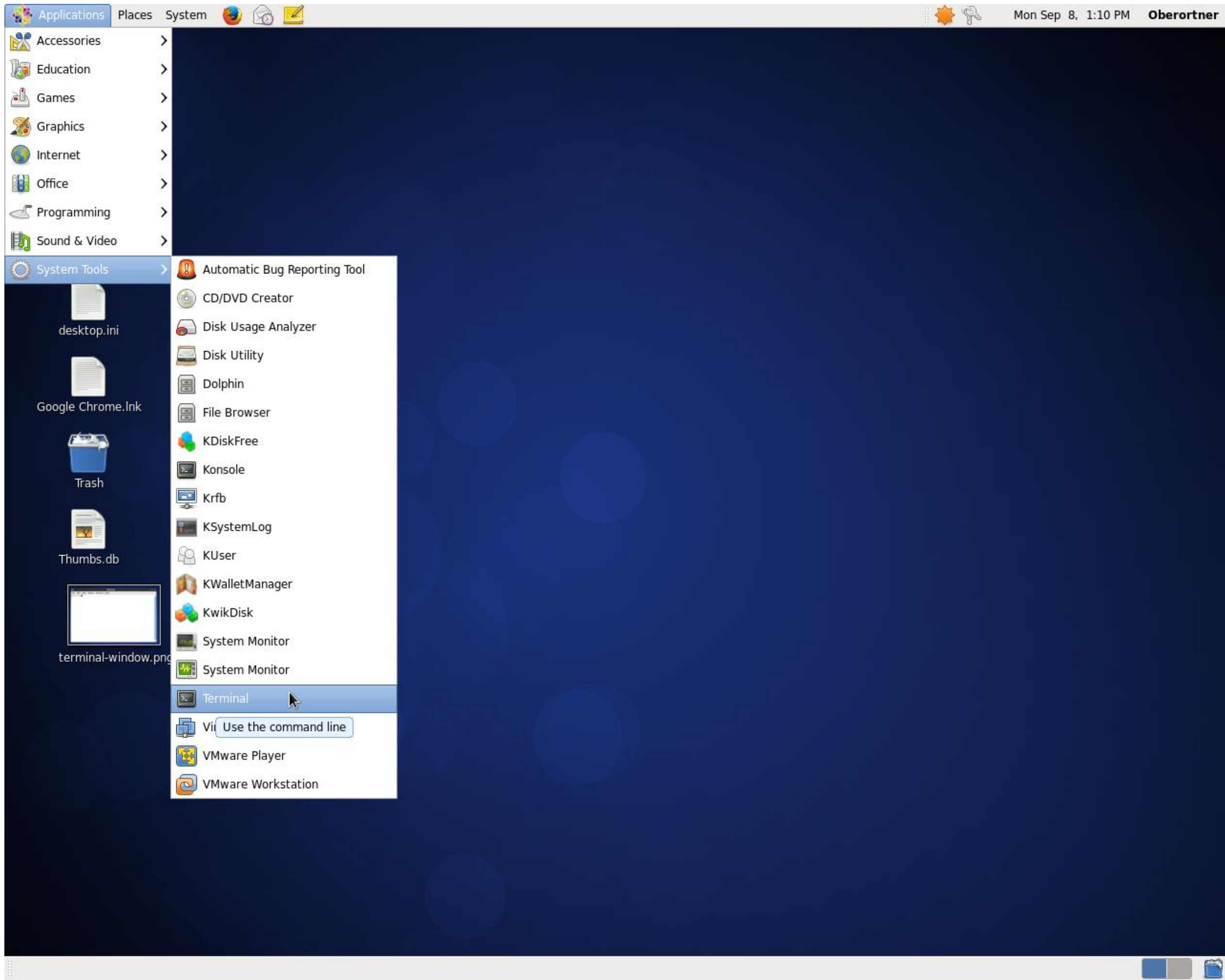
- **Introduction** to UNIX
- The **UNIX Shell**
- **Listing** Files and Directories
- **Navigating** through the File/Directory Tree
- Create, Copy, Move, Delete Files
- File **Editors**
- **Security and Access Rights**
- **“Man pages”**
- **Zipping** Files and Directories
- **Motivation/Purpose**
- “Little” **Assignment**

Introduction to UNIX

- Unix is a Computer Operating System (OS)
- Developed at AT&T Bell Labs
  - started in the late 60s
  - Ken Thompson and Dennis Ritchie
- Implemented in **C** and Assembler
- Prominent **Derivatives** of UNIX:
  - GNU/Linux
  - MacOS X
  - CentOS
- **Textual** UI: Shell
- **Graphical** UI: X Windows System

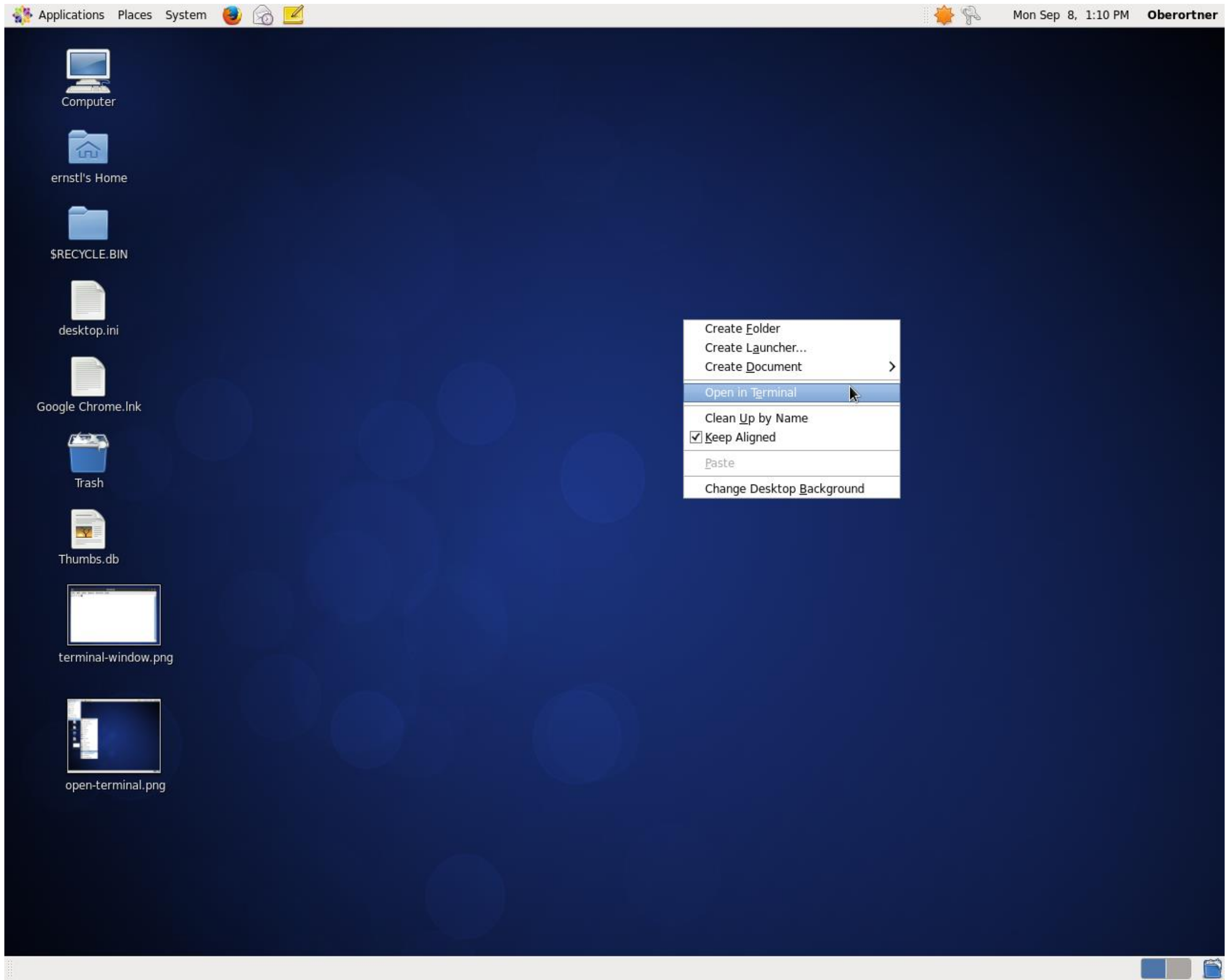
The UNIX Shell

- Textual UI to the OS Kernel
- Accessible via **Terminal** tool



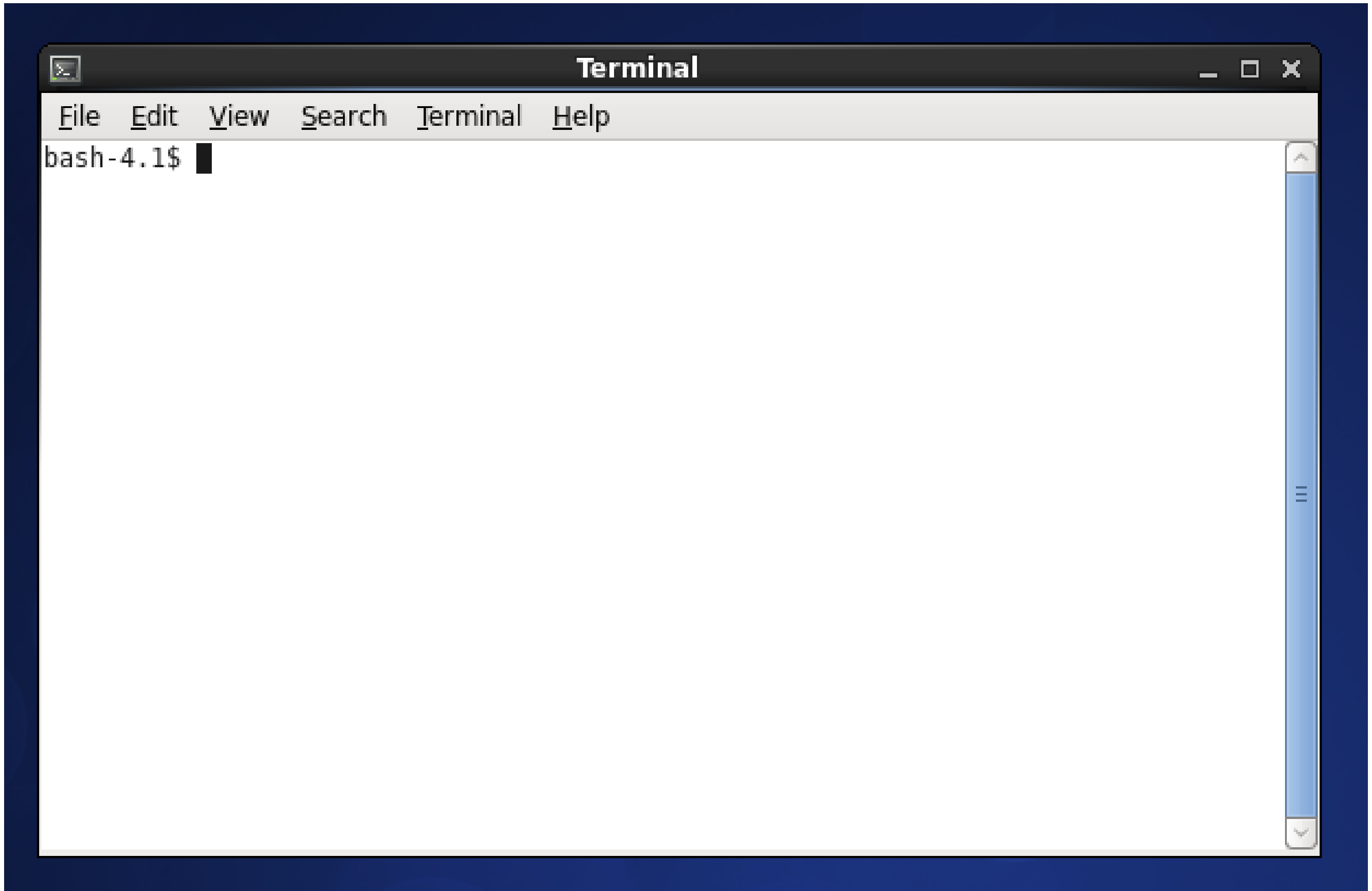
The UNIX Shell

- Textual UI to the OS Kernel
- Accessible via **Terminal** tool



The UNIX Shell

- Textual UI to the OS Kernel
- Accessible via **Terminal** tool



```
bash-4.1$ 
```

The UNIX Shell

- Textual UI to the OS Kernel
- Accessible via **Terminal** tool
- **Nice Features** of Terminal/Shell:
  - **History**: Up/down arrows
  - **Auto-completion**: Tab

Mostly every command takes **arguments**

```
<command> [<list-of-arguments>]*
```

Examples:

```
ls –l
cd ./my/directory
cp file1 file2
```

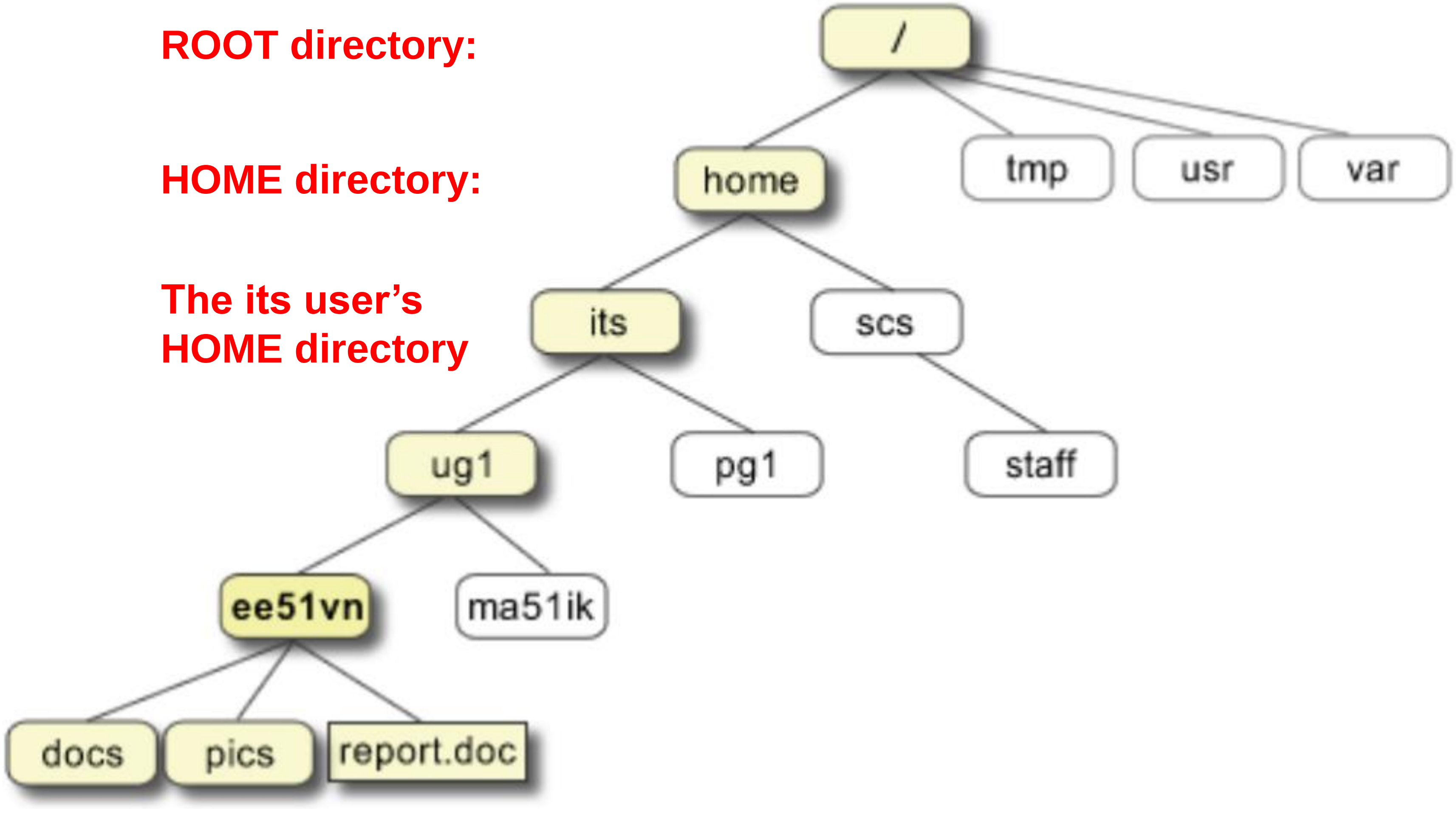
Files and Directories

The UNIX File Tree:

ROOT directory:

HOME directory:

The its user's HOME directory



Listing Files

- `ls` command **lis**t files and directories

- `ls -a`
  - list hidden files and directories
  - Names of hidden files and directories start with **.**

Navigating through Directories

- `cd` command **c**hange **d**irectory

`cd .` Go to the current directory

`cd ..` Go to the directory one level above

`cd ~` go to the HOME directory

`cd /absolute/path/to/directory`

ROOT Directory

`cd ./relative/path/to/directory`

Current Directory

- `pwd` command **p**rint **w**orking **d**irectory

Create, Copy, Move, Delete Files and Directories

- `touch` … create file

  Example: `touch main.cpp`
- `mkdir` … **make dir**ectory

Example: `mkdir EC327_Lab1`
- `cp` … **cop**y file

Example: `cp main.cpp main01.cpp`
- `mv` … **mov**e file

Example: `mv main.cpp main01.cpp`
- `rm` … **rem**ove file

Example: `rm main.cpp`

Read Files

- `less <filename>`
- Example: `less stuff.txt`
- Use space bar to go to next page
- Use “b” to go to previous page
- Press “q” to quit back to shell
- To show line numbers, use option –N:

  `less –N <filename>`

The vi Editor

Open vi: `vi` [<filename>]?

Two modes: EDIT (default) and **INSERT**

`i`	switch to **INSERT** mode
`ESC`	go back to EDIT mode

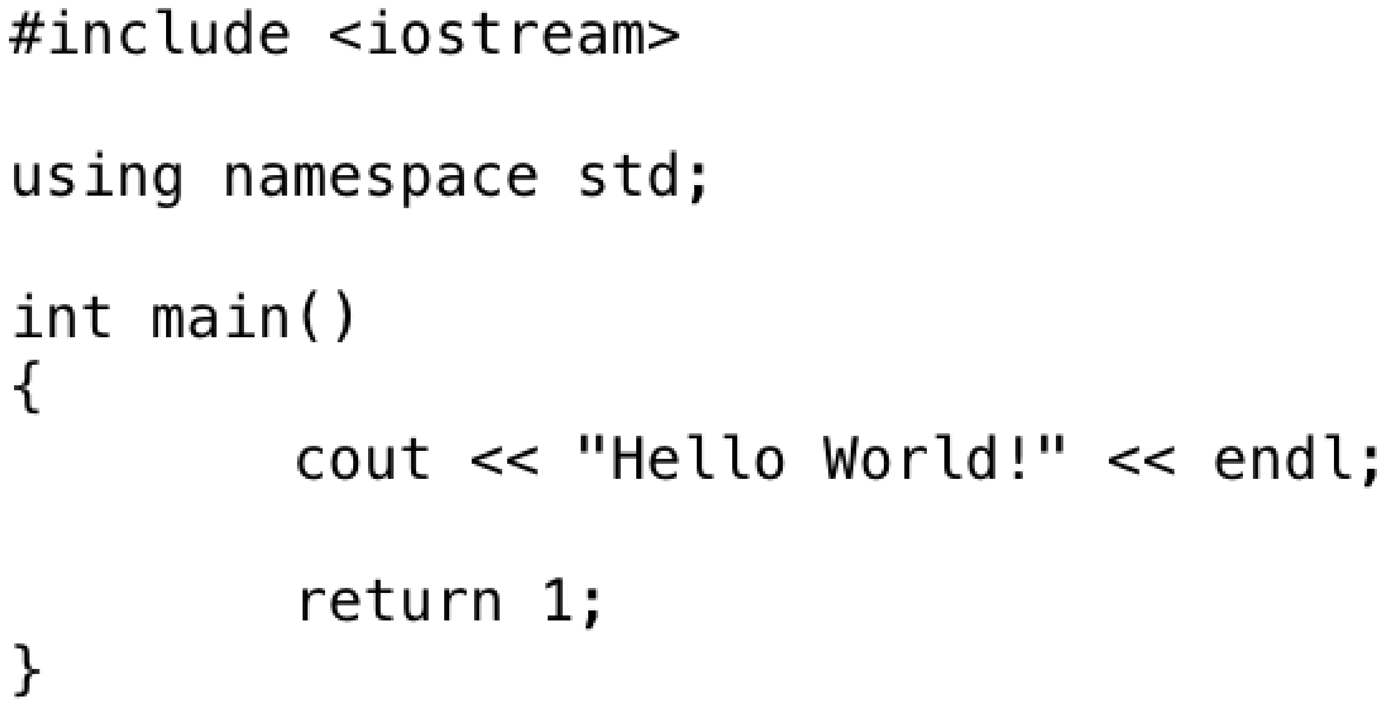
(some) EDIT mode commands:

`:q`	quit
`:q!`	force quit
`:wq`	write and quit
`dd`	delete line
`5G`	go to line #5 (capital G !)
`u`	undo

http://unixhelp.ed.ac.uk/vi/ref.html

Using the vi Editor

```
vi HelloWorld.cpp
```



```
#include <iostream>

using namespace std;

int main()
{
        cout << "Hello World!" << endl;

        return 1;
}
```

The emacs Editor

http://www.gnu.org/software/emacs/

Start emacs: `emacs` [<filename>]?

Start emacs to continue working in Shell:

`emacs` [<filename>]? **&**

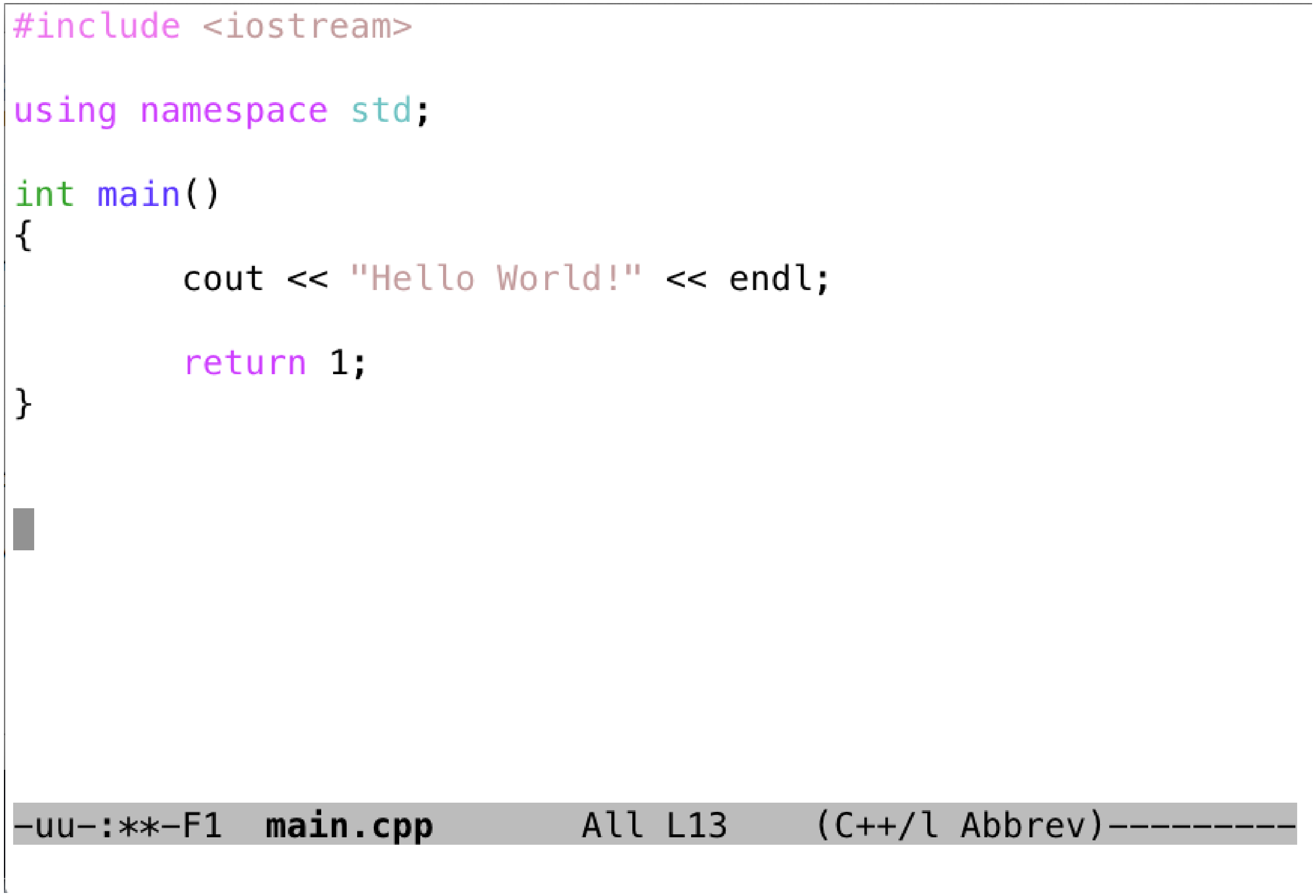
(some) emacs Commands:

^x^s	**Save File**
^x^c	**Quit**

& … start a process (e.g. the emacs Editor) in the background

Using the emacs Editor

```
emacs HelloWorld.cpp
```



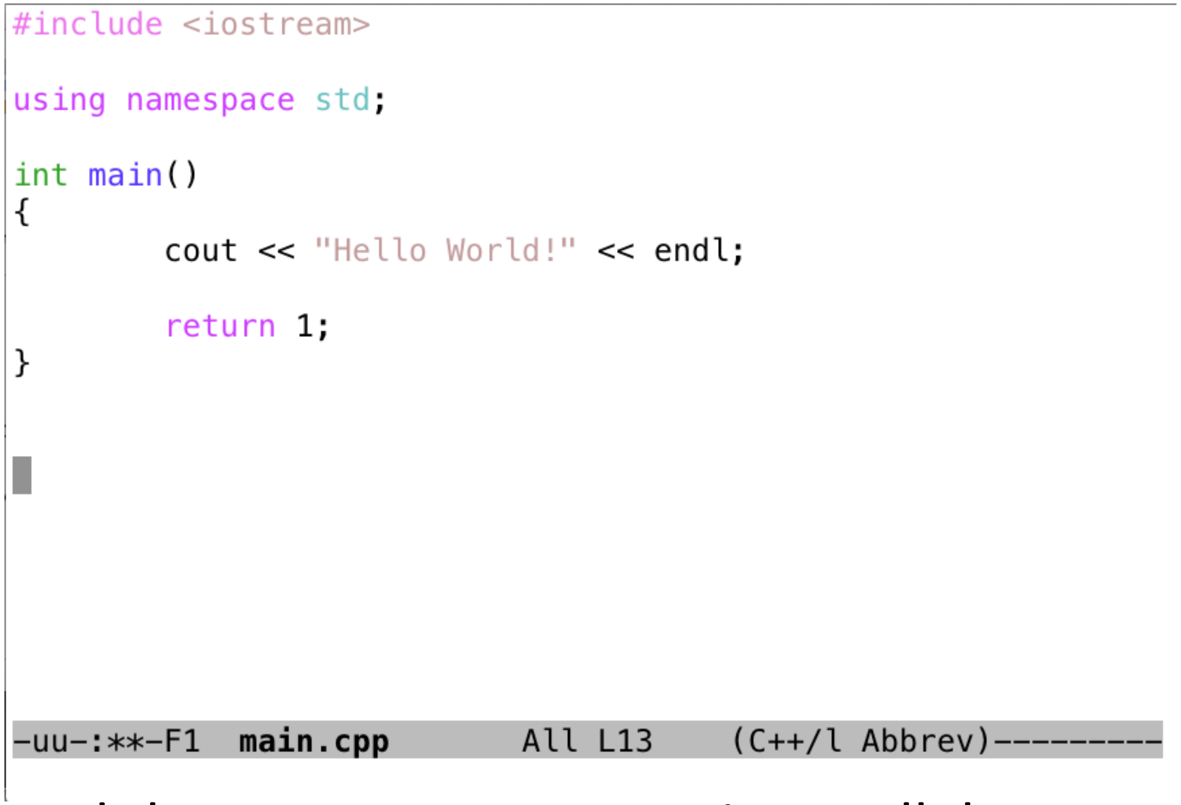
```
#include <iostream>

using namespace std;

int main()
{
        cout << "Hello World!" << endl;

        return 1;
}


-uu-:**-F1  main.cpp       All L13    (C++/l Abbrev)---------
```

^x^s … Save

^x^c … Quit

Running the emacs Editor in parallel with the Shell/Terminal

```
emacs HelloWorld.cpp &

ls -l
cd ..
```



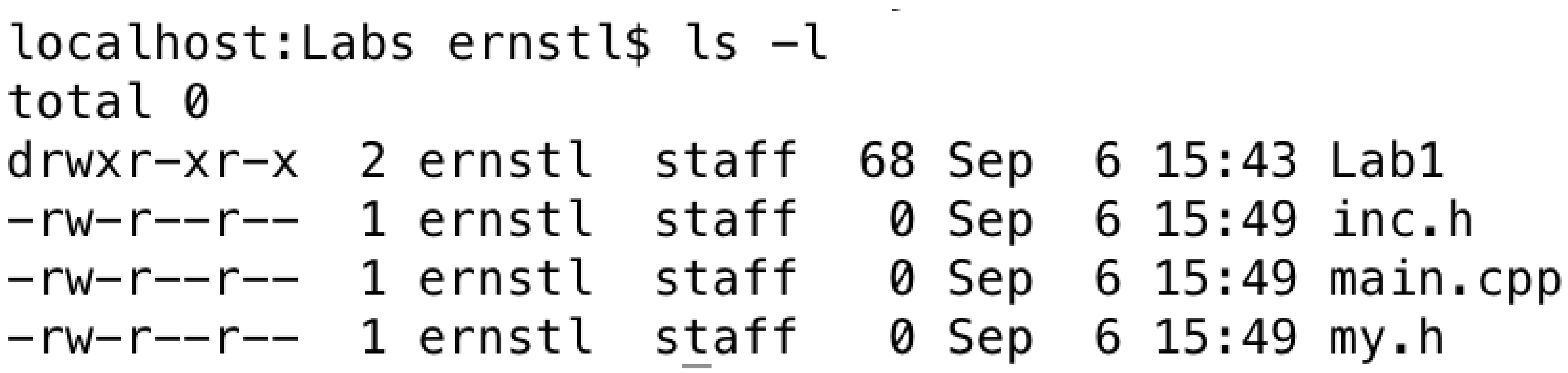
```
#include <iostream>

using namespace std;

int main()
{
	cout << "Hello World!" << endl;

	return 1;
}



-uu-:**-F1  main.cpp       All L13    (C++/l Abbrev)---------
```

The Shell/Terminal process and the emacs Process run in parallel.

Humans lose track easily when running multiple processes in parallel!

Security and Access Rights

`ls -l` list files in “long” format



```
localhost:Labs ernstl$ ls -l
total 0
drwxr-xr-x  2 ernstl  staff  68 Sep  6 15:43 Lab1
-rw-r--r--  1 ernstl  staff   0 Sep  6 15:49 inc.h
-rw-r--r--  1 ernstl  staff   0 Sep  6 15:49 main.cpp
-rw-r--r--  1 ernstl  staff   0 Sep  6 15:49 my.h
```

d	Directory
r	**r**ead
w	**w**rite
x	e**x**ecute

d r w x r – x r – x

user **group** **world**

Example:

user	**ernstl**
group	**staff**

Security and Access Rights

`chmod` **ch**ange file **mod**e

`chown` **ch**ange **own**er

We will not discuss those commands in more detail!

However, UNIX provides manuals!

“Man Pages”

- Man is an abbreviation for **Manual**
- `man <command>`
- Opens the manual in the **vi** Editor

Examples:

```
man ls
man cd
man g++
```

(Un)Zipping (1/2)

- Various compression formats and tools
  - tar
  - gzip
  - **zip** (used in EC327)
- The **zip** command and its arguments:

```
zip <zip-filename> <file>
zip -r <zip-filename> <directory>
```

Important!

While zipping a directory, you CANNOT place the zip-file into the directory you're zipping!

(Un)Zipping (2/2)

Example:

```
mkdir Lab1
cd Lab1
emacs main.cpp
~~zip –r Lab1.zip Lab1/*~~
```

Option 1:
Go to the directory above!

```
cd ..
zip –r Lab1.zip Lab1/*
```

Option 2:
Put the zip-file into the above directory!

```
zip –r ../Lab1.zip ../Lab1/*
```

Unzipping:

```
unzip <zip-filename>
```

Zipping
EC327 Midterm and Finals (1/2)

a. Open a terminal window.

b. Create a folder named **<yourBUusername>_<last4digitsofBUID>_<machineNumber>_final**. For example, student *James Bond* with BU username *jbond*, BU ID U*12340007*, and machine number *002* should type:

```
mkdir jbond_0007_002_final
```

*** Make sure you put in your own name and number here, and not those of James Bond's!!!*

c. Change directory to your final directory:

```
cd jbond_0007_002_final
```

d. Create `test.cpp` file in your directory, and write the following code:

```
#include <iostream>
using namespace std;
int main()
{
  cout<< "TEST EC327 FINAL" << endl;
  return 0;
}
```

Zipping
EC327 Midterm and Finals (2/2)

e. Zip your directory and submit via the following steps:

```
cd ..
ls
```

*** You should see your jbond_0007_002_final directory listed now.*

```
zip -r jbond_0007_002_final.zip ./jbond_0007_002_final/*
```

*** Make sure to replace James Bond's folder name with your folder name.*

f. Check that the zip file exists and contains all of the files:

```
ls
```

*** You should see jbond_0007_002_final.zip*

```
less jbond_0007_002_final.zip
```

*** This will list all the files in the zip. Make sure it contains all your files. ('q' to quit "less" program)*

COLLABORATIVE SUMMARY

- What do the following commands stand for, do, and what arguments do they take?

```
ls
cd
mv
man
:q
zip
ls -l
less
```

Resources

UNIX Tutorial for Beginners
http://www.ee.surrey.ac.uk/Teaching/Unix/

Learn UNIX in 10 Minutes
http://freeengineer.org/learnUNIXin10minutes.html

UNIX Tutorial (UC Berkeley)
http://people.ischool.berkeley.edu/~kevin/unix-tutorial/toc.html

Just use your favorite Internet Search Engine ☺

Motivation

- **Why using (and learning) UNIX?**
  - Family of OSs
  - Simple and small-scale **kernel**
  - Kernel provides **APIs to interact with hardware**
  - **C** is a language to **program and control systems and hardware**
  - …

Why Shell/Terminal?

- Efficient **Programming Languages are textual**

A computer program is an ordered set of instructions to be executed by a computer.

- “**Shell Scripting**”
  – Every command is a “simple” tool
  – Commands can be grouped together and executed in a desired order (**Workflow**)

Every Computer/Software Engineer should be familiar with UNIX and using the Shell!

QUESTIONS?

Staff
Instructors
Douglas Densmore (dougd@bu.edu, 358-6238, PHO 335)
Office hours: **Tuesdays and Thursdays noon-1pm**, also by appointment.

Teaching Fellows
Timothy Chong (ctimothy@bu.edu); Office Hours: **Fridays 1-3pm** (and by appt.)
Juilan Trinh (julest@bu.edu); Office Hours: **Wednesdays 5-7pm** (and by appt.)
Joshua Klein (joshuaahklein@gmail.com); Office Hours: **Fridays 2-4pm** (and by appt.)
Steve Wang (stevejw@bu.edu); Office Hours: **Thursdays 2-4pm** (and by appt.)
Allison Durkan (azulad7@bu.edu); Office Hours: **TBD** (and by appt.)
Aselya Aliyeva (aliyevaa@bu.edu); Office Hours: **Fridays 3-5pm** (and by appt.)