

ROMS/TOMS Beginners Training/Tutorial

The ROMS/TOMS beginners training/tutorial session will be held in room **G012B** on the ground floor of the **Red Centre** building on the University of New South Wales (UNSW) Kensington Campus. Upon entering the **centre wing**, walk down one flight of stairs. The training/tutorial will start at **12:30pm** with a short meet and greet to get everyone settled in. We will begin the training at **1pm**, ending at **4pm**. After the training we will have a happy hour just upstairs from the computer lab outside on the south side of Red Centre. The maps in this flier should help guide you to the tutorial location. We have also provided some Vi/UNIX reference commands.

Getting to UNSW from recommended hotels:

Car/Taxi Info:

UNSW is approximately 3.2km (1.8 miles) by road from Coogee Sands hotel. A walkable distance but if you wish to drive, head west on Dolphin Street. Take the first exit on the round about (Arden Street) then take the second right on to Coogee Bay Road. Make a sharp left on Perouse Road and take the first right on Soudan Street then the first left on Avoca Street. Take the first right on to Barker Street and follow it to **Gate 14** on your right.

Fees apply for on-campus parking which is sometimes in limited supply. Visitor timed parking is available on the **top level** of the Barker Street carpark (see **map 3**). After you park your car, find the ticket machine. The machine takes coins and will issue a ticket for the amount of time you pay for. You must leave the ticket in a visible place inside the front window of your car. Fees are \$6 for 5 hours up to \$13 for 12 hours. Alternate parking may also be available on the **top level** of the Botany Street carpark (enter the University at **Gate 11** from Botany Street, also **map 3**).

Once parked and paid for, use **map 3** to help you find the Red Centre building; the training session is in room **G012B** on the ground floor of the centre wing. After entering the **centre wing**, walk down one flight of stairs.

Walking:

The other hotels are quite close to UNSW and it is recommended that you walk since parking is limited and fees apply. **Maps 1 & 2** should help you get to the UNSW campus and **Map 3** will help you find the **Red Centre** building; the training session is in room **G012B** on the ground floor of the centre wing. After entering the **centre wing**, walk down one flight of stairs.

Getting to UNSW from Sydney Airport:

Car/Taxi Info:

UNSW is approximately 10km by road from Sydney Airport. Enter the University at **Gate 14** from Barker Street (**map 3**). Fees apply for on-campus parking which is sometimes in limited supply. Visitor timed parking is available on the **top level** of the Barker Street carpark (see **map 3**). After you park your car, find the ticket machine. The machine takes coins and will issue a ticket for the amount of time you pay for. You must leave the ticket in a visible place inside the front window of your car. Fees are \$6 for 5 hours up to \$13 for 12 hours. Alternate parking may also be available on the **top level** of the Botany Street carpark (enter the University at **Gate 11** from Botany Street, also **map 3**).

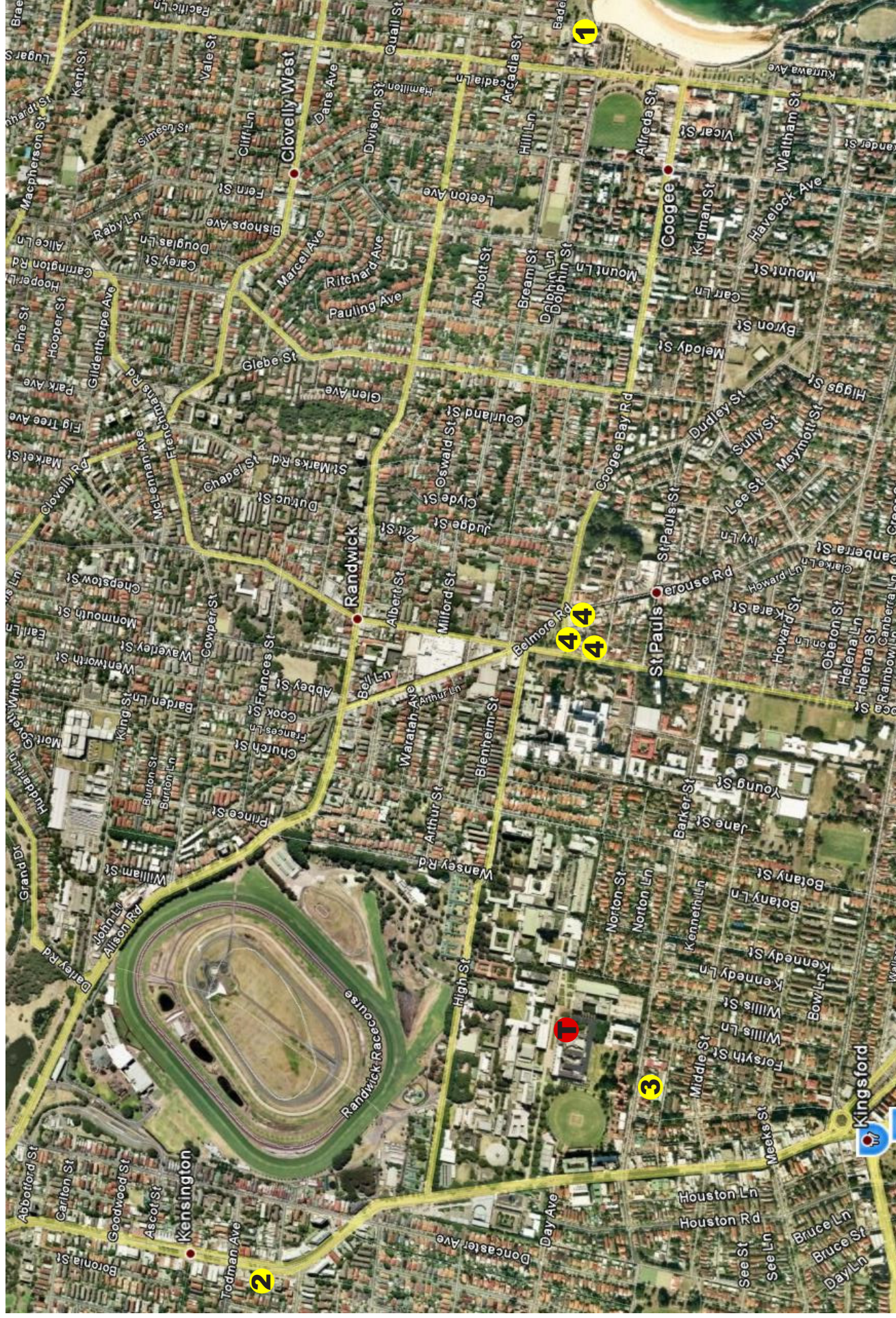
Once parked and paid for, use **map 3** to help you find the Red Centre building; the training session is in room **G012B** on the ground floor of the centre wing. After entering the **centre wing**, walk down one flight of stairs.

Bus Info:

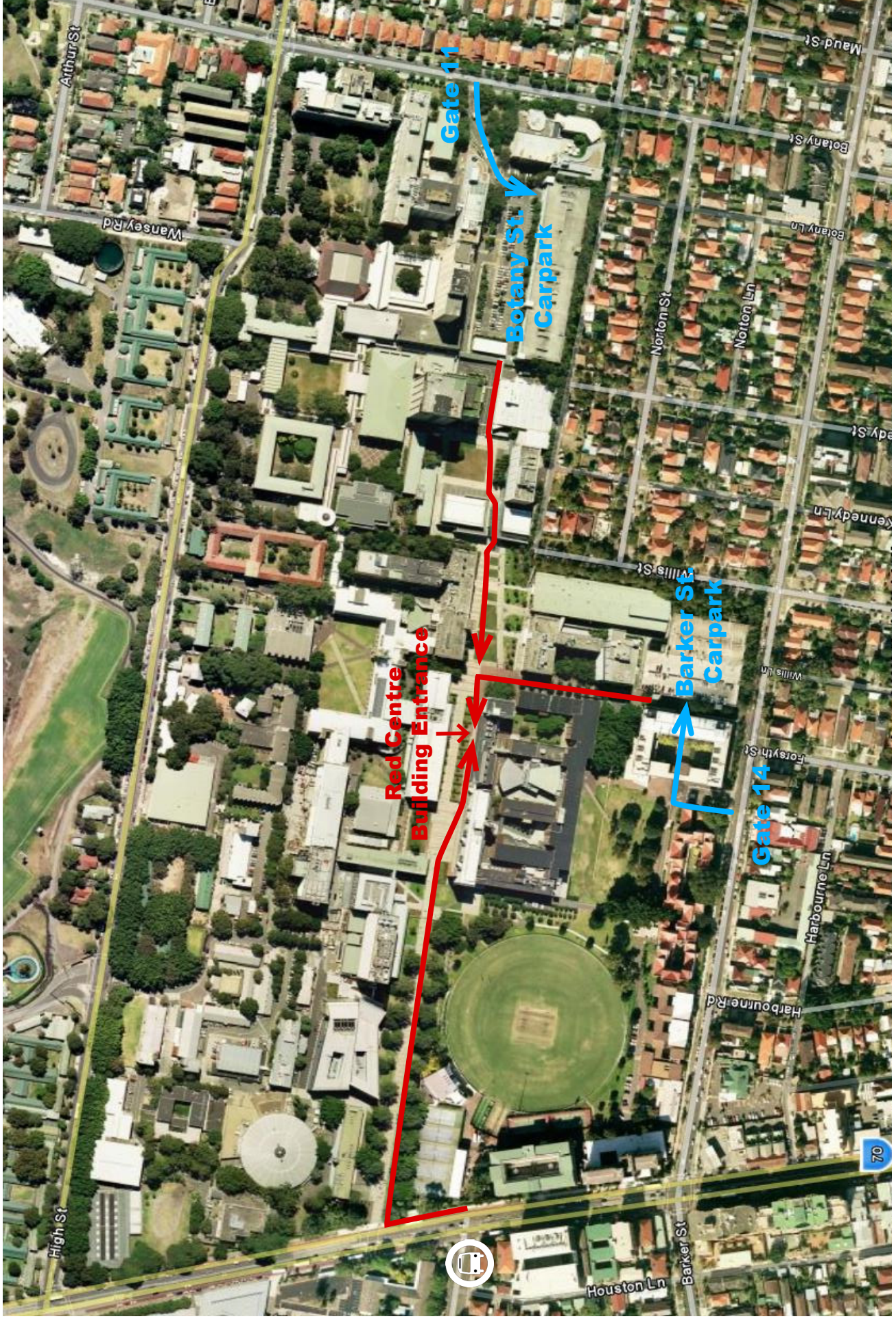
Take the **400** bus from Sydney Airport, alight at the bus stop opposite UNSW near the intersection of **Day Avenue** with **Anzac Parade**. The bus stop is marked on **map 3**, which will help you find your way to the Red Centre building; the training session is in room **G012B** on the ground floor of the centre wing. After entering the **centre wing**, walk down one flight of stairs.

Hotel Location Map 1: Hotels (yellow) and Tutorial location (red)

1. Coogee Sands Hotel 2. Addison Hotel 3. Barker Lodge 4. The Randwick Lodges  Red Centre Building



Map 3: Parking (blue) and walking routes (red) to tutorial location



Vi Reference Sheet

Modes

Vi has two modes: insertion mode, and command mode. The editor begins in command mode, where cursor movement and text deletion and pasting occur. Insertion mode begins upon entering an insertion or change command. [ESC] returns the editor to command mode (where you can quit, for example by typing :q!). Most commands execute as soon as you type them except for “colon” commands which execute when you press the return key.

Quitting

exit, saving changes	:x
quit (unless changes)	:q
quit (force, even if unsaved)	:q!

Inserting text

insert before cursor, before line	i , I
append after cursor, after line	a , A
open new line after, line before	o , O
replace one char, many chars	r , R

Deleting text

Almost all deletion commands are performed by typing d followed by a motion. For example dw deletes a word. A few other deletions are:

character to right, left	x , X
to end of line	D
line	dd
line	:d

Yanking text

Like deletion, almost all yank commands are performed by typing y followed by a motion. For example y\$ yanks to the end of line. Two other yank commands are:

line	yy
line	:y

Changing text

The change command is a deletion command that leaves the editor in insert mode. It is performed by typing c followed by a motion. For example cw changes a word. A few other change commands are:

to end of line	C
line	cc

Putting text

put after position of after line	p
put before position of before line	P

Search for strings

search forward	/string
search backward	?string
repeat search in same, reverse directions	n , N

Replace

The search and replace function is accomplished with the :s command. It is commonly used in combination with ranges or the :g command (below):

replace pattern with string	s/pattern/string/flags
flags: all on each line, confirm each	g , c
repeat last :s command	&

Regular (pattern) expressions

any single character except newline	. (dot)
zero or more repeats	*
any character in set	[...]
any character not in set	[^...]
beginning, end of line	^ , \$
beginning, end of word	\< , \>
grouping	\(. . . \)
contents of n th grouping	\n

Ranges

Ranges may precede most “colon” commands and cause them to be executed on a line or lines. For example :3,7d would delete lines 3–7. Ranges are commonly combined with the :s command to perform a replacement on several lines, as with :.,\$s/pattern/string/g to make a replacement from the current line to the end of the file.

lines n-m	:n,m
current line	:.
last line	:\$
all lines	:%
all matching lines	:g/pattern/

Files

write file (current file if not name given)	:w file
append file (current file if no name given)	:w >>file
read file after line	:r file
next file	:n
previous file	:p
edit new file	:e file

Bash Reference Sheet

File Commands

directory listing	ls
formatted listing with hidden files	ls -al
change directory to <i>dir</i>	cd <i>dir</i>
change to home	cd
show current directory	pwd
create directory <i>dir</i>	mkdir <i>dir</i>
delete <i>file</i>	rm <i>file</i>
delete directory <i>dir</i>	rm -r <i>dir</i>
force remove <i>file</i>	rm -f <i>file</i>
force remove directory <i>dir</i>	rm -rf <i>dir</i>
copy <i>file1</i> to <i>file2</i>	cp <i>file1 file2</i>
copy <i>dir1</i> to <i>dir2</i> ; create <i>dir2</i> if it doesn't exist	cp -r <i>dir1 dir2</i>
rename/move <i>file1</i> to <i>file2</i> . If <i>file2</i> is existing directory, moves <i>file1</i> into directory <i>file2</i>	mv <i>file1 file2</i>
create symbolic link <i>link</i> to <i>file</i>	ln -s <i>file link</i>
create or update <i>file</i>	touch <i>file</i>
places standard input into <i>file</i>	cat > <i>file</i>
output contents of <i>file</i>	more <i>file</i>
output the first 10 lines of <i>file</i>	head <i>file</i>
output the last 10 lines of <i>file</i>	tail <i>file</i>
output the contents of file as it grows, starting with the last 10 lines	tail -f <i>file</i>

Process Management

display your currently active processes	ps
display all running processes	top
kill process id <i>pid</i>	kill <i>pid</i>
kill all precesses named <i>proc</i>	kill <i>proc</i>
lists stopped or background jobs; resume a stopped job in the background	bg
brings the most recent job to foreground	fg
brings job <i>n</i> to the foreground	fg <i>n</i>

File Permissions

chmod octal file – change the permissions of *file* to *octal*, which can be found separately for user, group, and world by adding:

- 4 – read (r)
- 2 – write (w)
- 1 – execute (x)

Examples:

chmod 777 – read, write, execute for all

chmod 755 – rwx for owner, rx for group and world

For more options, see **man chmod**.

SSH

connect to <i>host</i> as <i>user</i>	ssh <i>user@host</i>
---------------------------------------	-----------------------------

Searching

search for <i>pat</i> in <i>files</i>	grep <i>pat files</i>
search recursively for <i>pat</i> in <i>dir</i>	grep -r <i>pat dir</i>
search for <i>pat</i> in output of <i>com</i>	<i>com</i> grep <i>pat</i>
find all instances of <i>file</i>	locate <i>file</i>

System Info

show the current date and time	date
show kernel information	uname -a
show the manual for <i>command</i>	man <i>command</i>
show disk usage	df
show directory space usage	du
show possible locations of <i>app</i>	whereis <i>app</i>
show which <i>app</i> will be run by default	which <i>app</i>

Compression

create a tar named <i>file.tar</i> containing <i>files</i>	tar cf <i>file.tar files</i>
extract the files from <i>file.tar</i>	tar xf <i>file.tar</i>
create tar named <i>file.tar.gz</i> containing <i>files</i> with Gzip compression	tar czf <i>file.tar.gz files</i>
extract a tar using Gzip	tar xzf <i>file.tar.gz</i>
create a tar with Bzip2 compression	tar cjf <i>file.tar.bz2 files</i>
extract a tar using Bzip2	tar xjf <i>file.tar.bz2</i>
compress <i>file</i> and renames it to <i>file.gz</i>	gzip <i>file</i>
decompress <i>file.gz</i> back to <i>file</i>	gzip -d <i>file.gz</i>

Network

ping <i>host</i> and output results	ping <i>host</i>
download <i>file</i>	wget <i>file</i>
continue a stopped download	wget -c <i>file</i>

Shortcuts

halt the current command	Ctrl+C
Suspends the current command	Ctrl+Z
erase current line	Ctrl+U
show previous commands	Up arrow
repeat the last command	!!
log out of currnet session	exit

Environment (bash)

show all current evironment variables and values	printenv
show current value of environment variable <i>VAR</i>	printenv <i>VAR</i>
set environment variable <i>VAR</i> to <i>value</i>	export <i>VAR=value</i>