Warnings

First of all, these notes will cover only a small subset of the available commands and utilities, and will cover most of those in a shallow fashion.

Read the relevant material in Sobell!

If you want to follow along with the examples that follow, and you do, open a Linux terminal.

Second, most of the Linux commands have features that are enabled by using command-line switches; check the **man** pages for the commands for details!

Getting Started

The Linux terminal (or command shell) allows you to enter commands and execute programs.

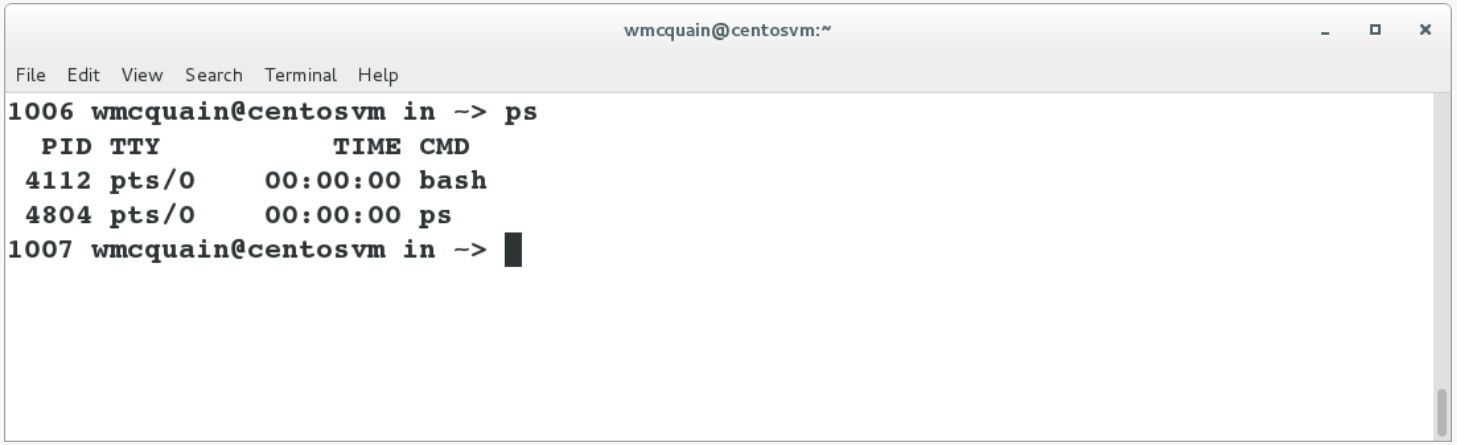
A terminal displays a prompt and a cursor indicating it's waiting for user input:

The prompt itself is configurable; precisely how depends on the particular type of shell you are running.

It is likely that by default you will run the *bash* shell.

What's Running?

The **ps** command displays information about processes the shell is currently running:



We see that two processes are executing, **bash** and **ps**.

Moreover, we see that:

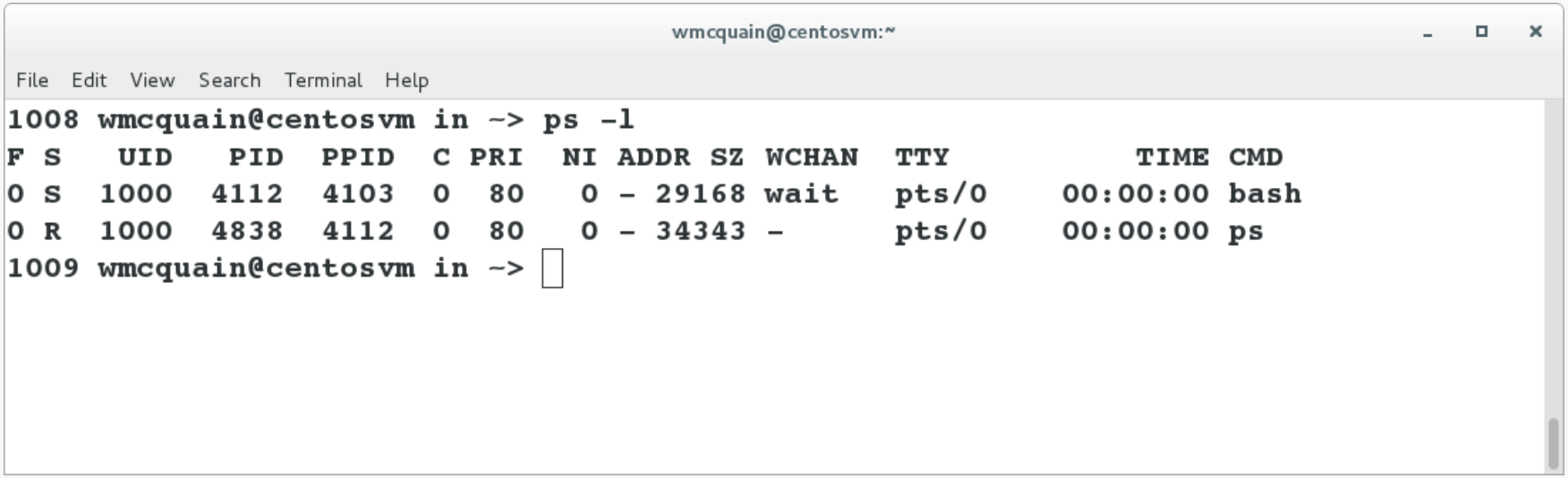
- each is assigned a unique numeric identifier called a process ID or PID
- each is associated with a terminal (TTY) named pts/0

Try executing **ps** a second time… you'll notice that the PID for **bash** is the same as before but the PID for **ps** has changed.

Why? (That's two questions.)

More Information

Try running **ps** with the **–l** (that's ell, not one) switch:

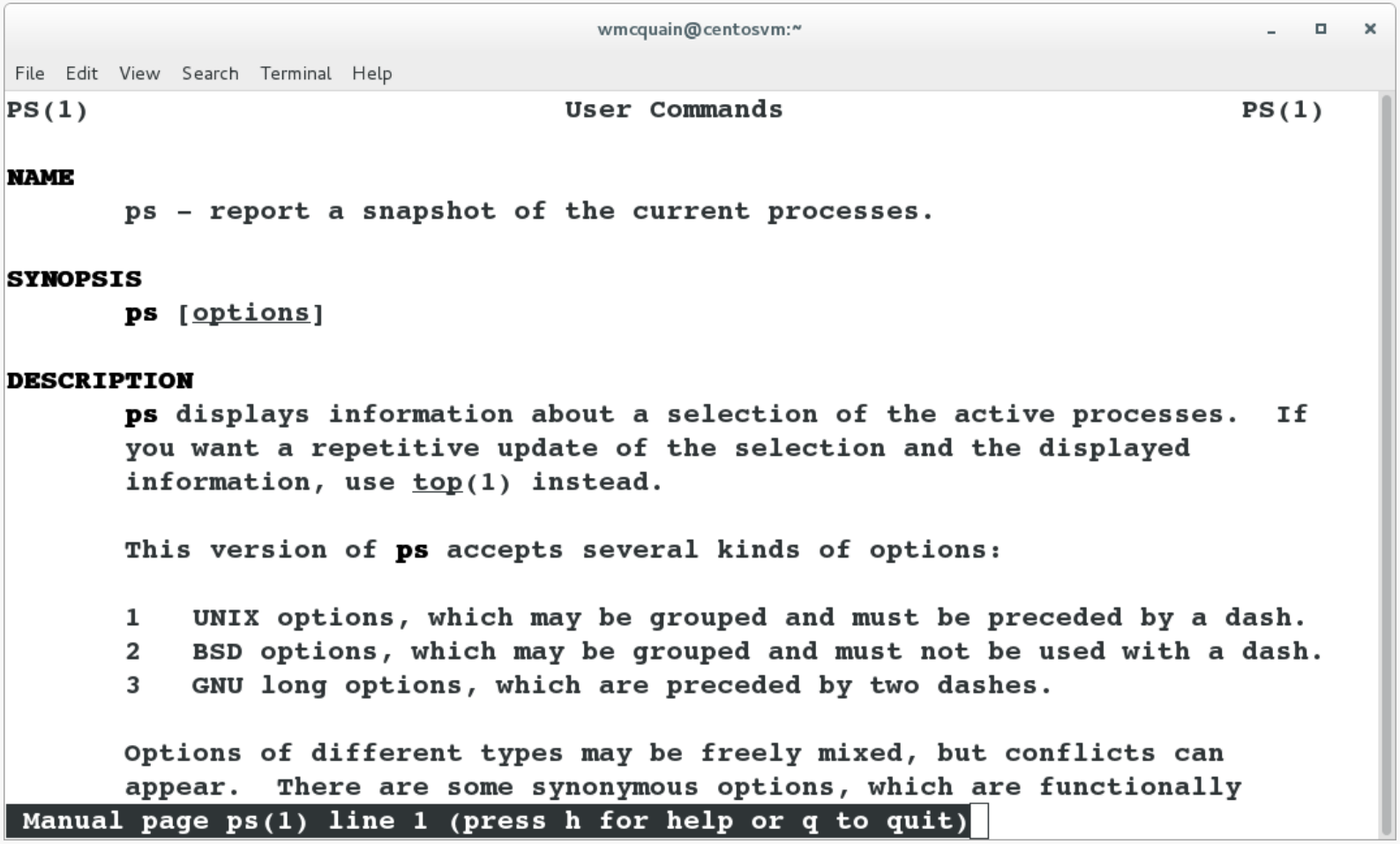


Don't worry about the meaning of all that just yet, but do notice that the results of the **ps** command were altered by the use of a "switch" on the command line.

This is typical of Linux commands and many user programs.

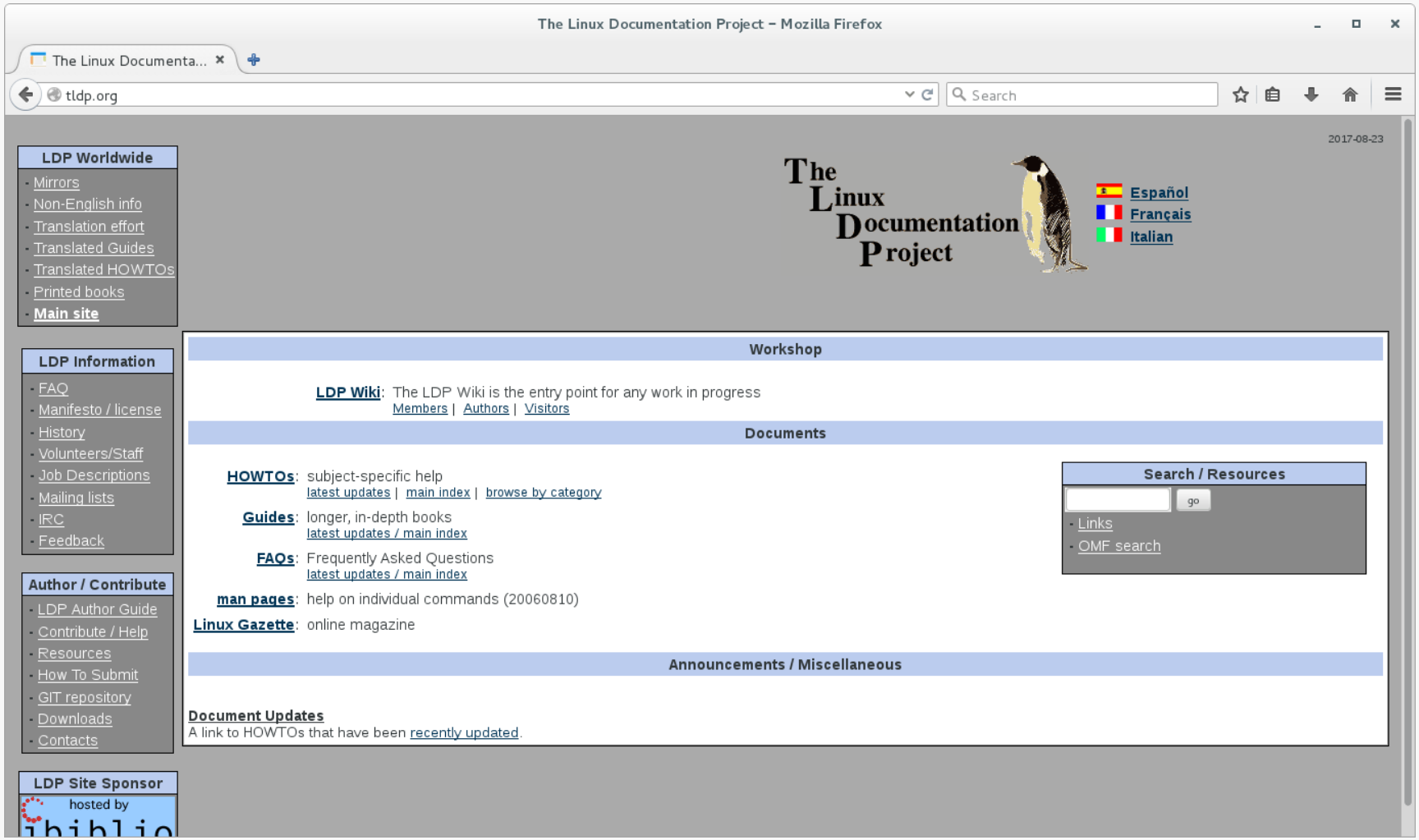
One way to find out more…

The **man** (manual) command can be used to obtain more information about Linux commands:



The **man** pages are often the first resort for discovering options.
Try running **man man**…

The Linux Documentation Project

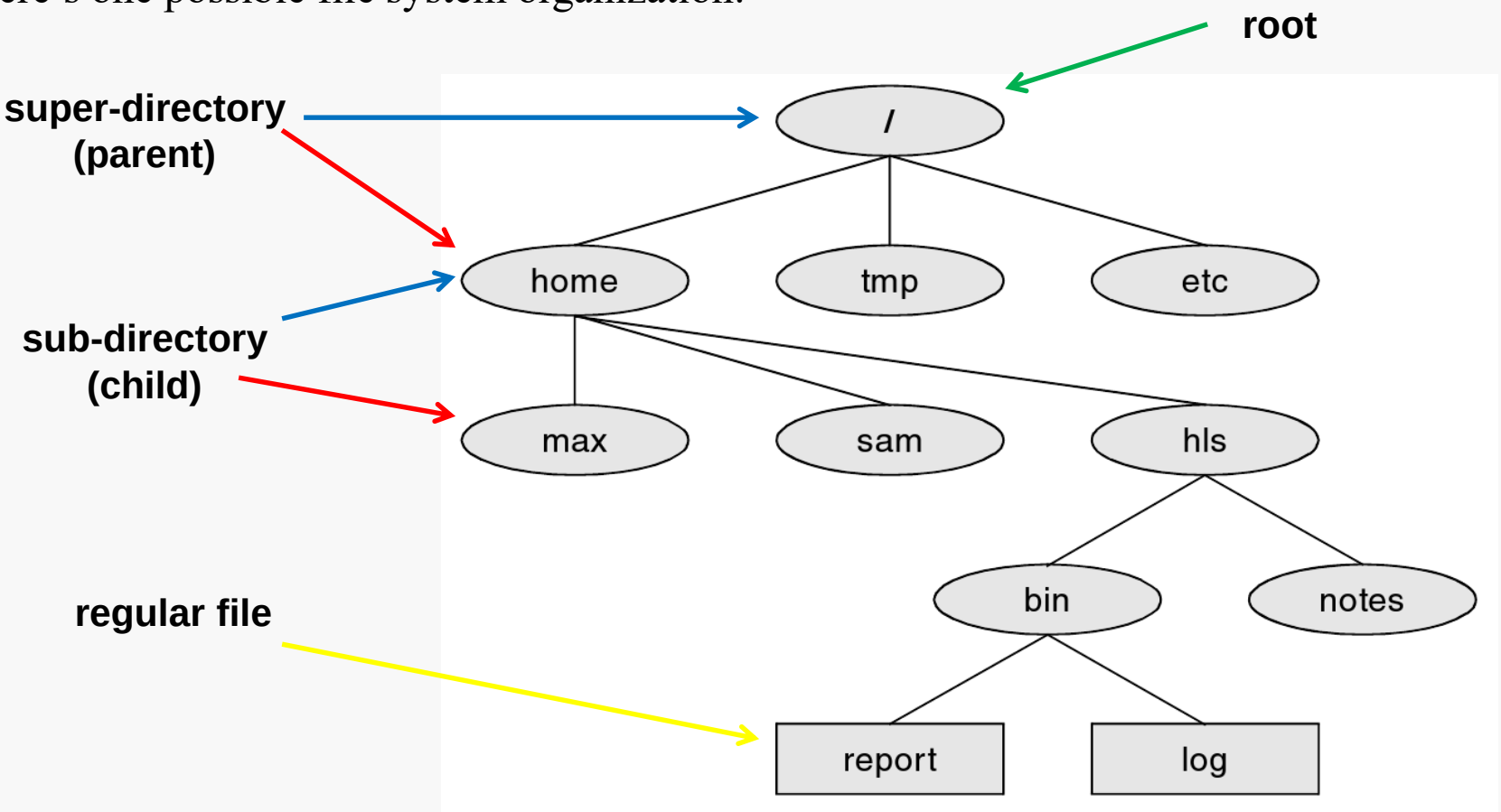


File System Basics

The file system is a set of data structures that organizes collections of files.

Files are grouped into *directories* (although directories are themselves files).

Here's one possible file system organization:



File Names

Each file and directory has a name:

- names are case-sensitive
- names within the same directory must be unique
- the use of characters other than letters, digits, underscores and periods tends to cause extra work when using many commands

File names are often of the form <name.ext>, such as `BashTerminal.jpg`.

While file extensions are not mandatory, it is standard (and good) practice to employ them.

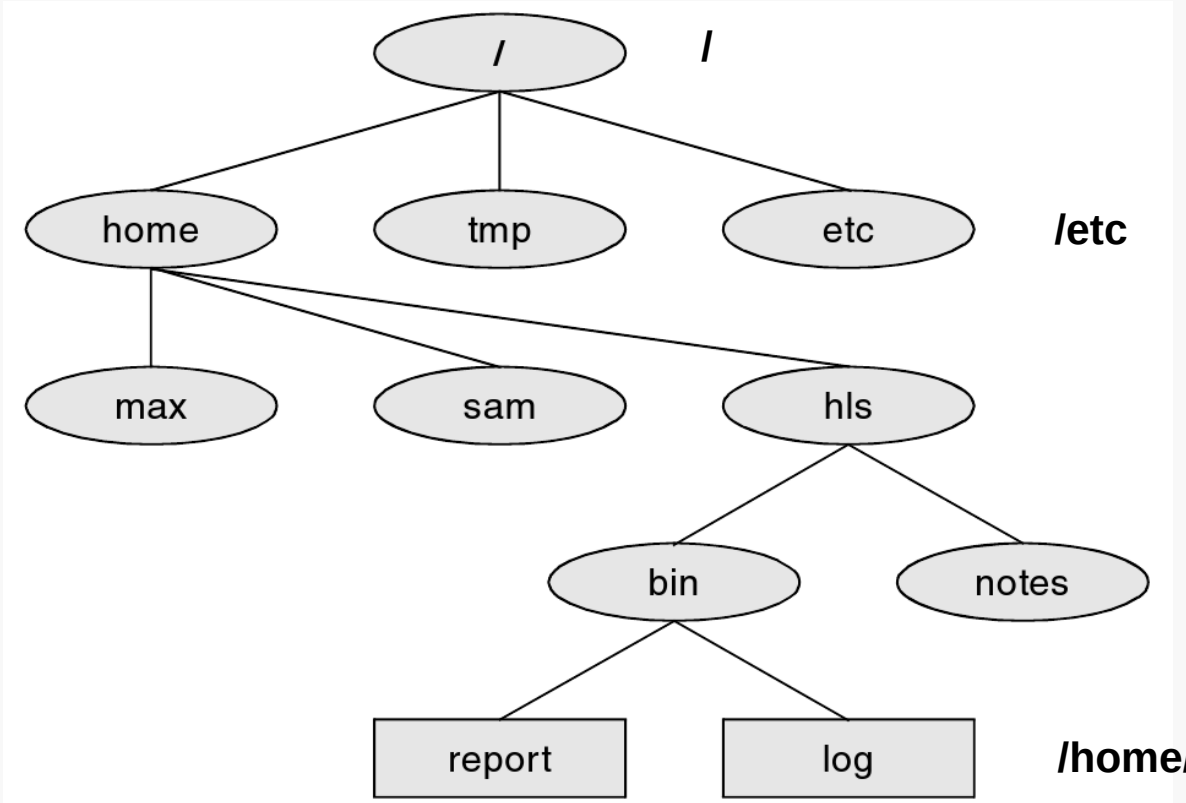
You are required to use appropriate file extensions in this course.

It is bad practice to employ extensions incorrectly. Common ones include:

- `c` C language source files
- `h` C language header files
- `txt` plain text files
- `gz` file compressed with **gzip**
- `tar` archive file created with **tar**
- `html` hypertext markup language file

Absolute Pathnames

Each file (and directory) can be specified by a unique *absolute pathname*:



Home Directory

When you open a terminal, by default you will be in your *home directory*.

Typically, this will be /home/<userid>, but you can check the path to your current directory by using the **pwd** command:

What's in here?

The **ls** command lists the files in the current directory:

wmcquain@centosvm:~/bin

File Edit View Search Terminal Help

```
1036 wmcquain@centosvm in ~/bin> ls
2505Summer16.jar    distributetars.sh  logisim-generic-2.7.1.jar  unpackjars.sh
3114Summer2016.jar  filter.sh          Mars4_4.jar                unpacktars2.sh
backtoshare.sh      flatten.sh         Mars4_5.jar                unpacktars.sh
backup.sh           Harvester.jar      package.sh                 unpackzips.sh
distributefiles.sh  howmany.sh         rmdupes3.sh                zipdirs.sh
1037 wmcquain@centosvm in ~/bin> ls -l
total 14656
-rwxrw----. 1 wmcquain comporg   308335 Jul  1  2016 2505Summer16.jar
-rwxrw----. 1 wmcquain comporg   308335 Jul  1  2016 3114Summer2016.jar
-rwxrwx---. 1 wmcquain comporg     2598 Jul  5 11:03 backtoshare.sh
-rwxrw----. 1 wmcquain wmcquain    2786 Jun 29  2015 backup.sh
-rwxrwx---. 1 wmcquain wmcquain    1929 Feb  6  2016 distributefiles.sh
-rwxr-----. 1 wmcquain wmcquain    2596 Nov 18  2014 distributetars.sh
-rwxrw----. 1 wmcquain wmcquain    6350 May  1 11:21 filter.sh
-rwxrw----. 1 wmcquain wmcquain    1968 Feb  4  2017 flatten.sh
-rwxrw----. 1 wmcquain comporg     3487 Jun  9 13:54 Harvester.jar
-rwxrw----. 1 wmcquain wmcquain    1380 Jun 29  2015 howmany.sh
-rwxrwx---. 1 wmcquain wmcquain 6933898 Jan 31  2017 logisim-generic-2.7.1.jar
-rwxrwx---. 1 wmcquain wmcquain 3212322 Nov 10  2014 Mars4_4.jar
-rwxrwx---. 1 wmcquain wmcquain 4169142 Aug 13  2015 Mars4_5.jar
-rwxrwx---. 1 wmcquain comporg     1502 Aug  4 22:41 package.sh
-rwxrw----. 1 wmcquain wmcquain    2449 Feb 23  2014 rmdupes3.sh
-rwxrwx---. 1 wmcquain comporg     2918 Jun 25  2016 unpackjars.sh
```

Directory Tree

You can display a map of the directory tree rooted at your current directory:

wmcquain@centosvm:~/2505

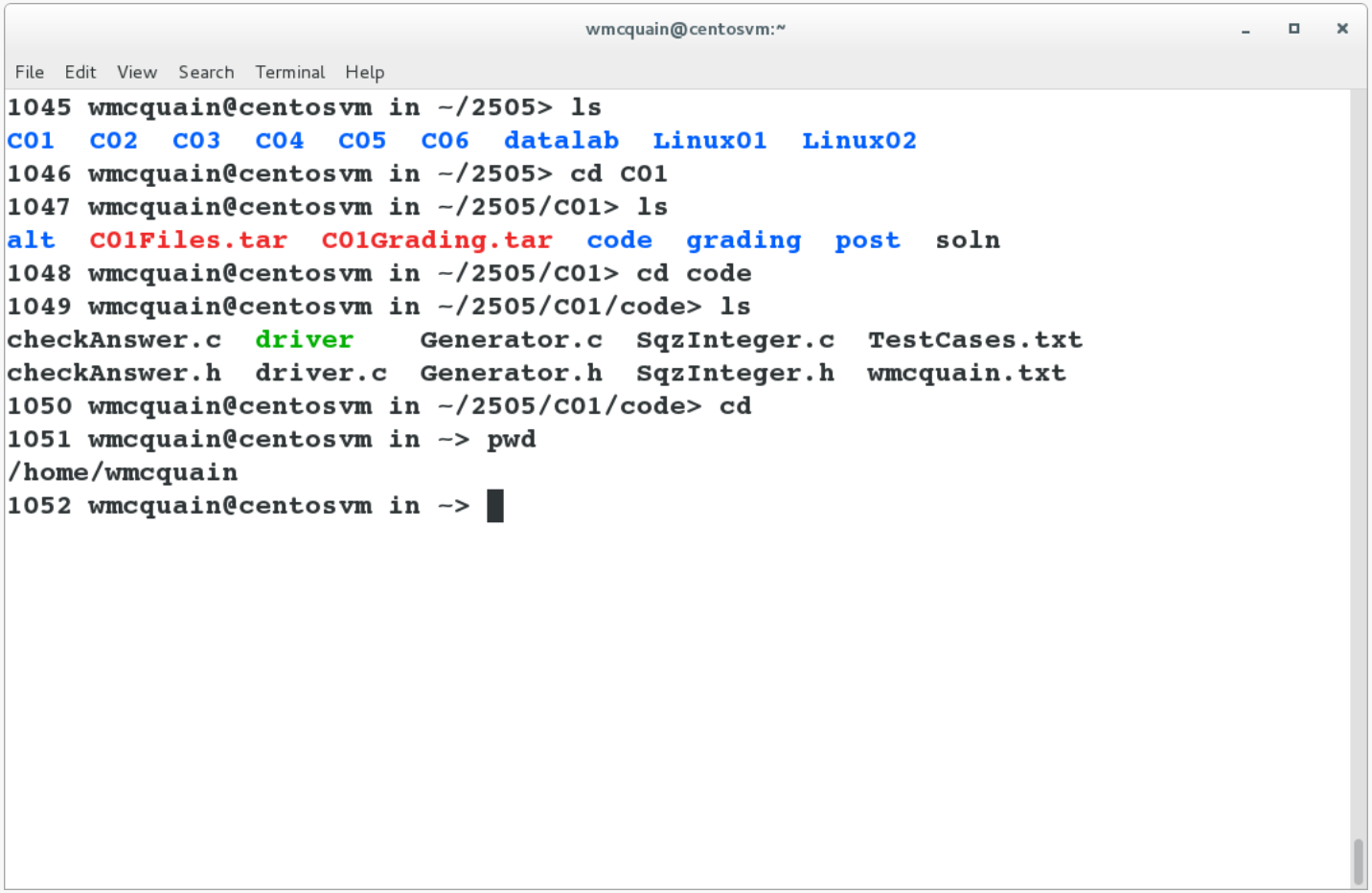
File Edit View Search Terminal Help

```
1040 wmcquain@centosvm in ~/2505> ls
C01  C02  C03  C04  C05  C06  datalab  Linux01  Linux02
1041 wmcquain@centosvm in ~/2505> tree
.
|-- C01
|   |-- alt
|   |   |-- altdriver
|   |   |-- checkAnswer.h
|   |   |-- checkAnswer.o
|   |   |-- driver.c
|   |   |-- SqzInteger.c
|   |   `-- SqzInteger.h
|   |-- C01Files.tar
|   |-- C01Grading.tar
|   |-- code
|   |   |-- checkAnswer.c
|   |   |-- checkAnswer.h
|   |   |-- driver
|   |   |-- driver.c
|   |   |-- Generator.c
|   |   |-- Generator.h
|   |   |-- SqzInteger.c
|   |   |-- SqzInteger.h
|   |   |-- TestCases.txt
```

The tree program may not be installed by default; we'll cover software package installation a bit later.

Directory Navigation

You can use the **cd** command to change your current (or *working*) directory:



Using **cd** with no destination moves you back to your home directory:

Relative Pathnames

You can also specify a pathname that's relative to the current (working) directory.

Let's say you're in a directory `~/2505/C01/alt/`:

```
wmcquain@
File  Edit  View  Search  Terminal  Help
1040 wmcquain@centosvm in ~/2505> ls
C01  C02  C03  C04  C05  C06  datalab
1041 wmcquain@centosvm in ~/2505> tree
.
|-- C01
|   |-- alt
|   |   |-- altdriver
|   |   |-- checkAnswer.h
|   |   |-- checkAnswer.o
|   |   |-- driver.c
|   |   |-- SqzInteger.c
|   |   `-- SqzInteger.h
|   |-- C01Files.tar
|   |-- C01Grading.tar
|   |-- code
|   |   |-- checkAnswer.c
|   |   |-- checkAnswer.h
|   |   |-- driver
|   |   |-- driver.c
|   |   |-- Generator.c
|   |   |-- Generator.h
|   |   |-- SqzInteger.c
|   |   |-- SqzInteger.h
|   |   |-- TestCases.txt
```

../C01Grading.tar

../code/checkAnswer.c

There are two special directory names:

- `.` refers to the current directory
- `..` refers to the parent of the current directory

Making/Removing a Directory

mkdir: creates a new subdirectory of the current directory

rmdir: deletes a empty subdirectory

rm -Rf: deletes a subdirectory and all its contents (recursive, very dangerous!)

```
wmcquain@centosvm:~/2505/notes
File  Edit  View  Search  Terminal  Help
1063 wmcquain@centosvm in ~/2505/notes> ls
T02
1064 wmcquain@centosvm in ~/2505/notes> mkdir temp
1065 wmcquain@centosvm in ~/2505/notes> ls
T02  temp
1066 wmcquain@centosvm in ~/2505/notes> ls temp
1067 wmcquain@centosvm in ~/2505/notes> rmdir temp
1068 wmcquain@centosvm in ~/2505/notes> ls
T02
1069 wmcquain@centosvm in ~/2505/notes> ls T02
Slide02.png  Slide04.png  Slide06.png  Slide11.png  Slide13.png
Slide03.png  Slide05.png  Slide10.png  Slide12.png
1070 wmcquain@centosvm in ~/2505/notes> rmdir T02/
rmdir: failed to remove 'T02/': Directory not empty
1071 wmcquain@centosvm in ~/2505/notes>
```

Copying Files: cp

You can create a copy of a file with the **cp** command.

Assume we're in a directory containing a file named **infloop.c**:

cp infloop.c infloop2.c

makes a copy of **infloop.c** named **infloop2.c** in the same directory

cp infloop.c ..

makes a copy of **infloop.c** with the same name in the parent directory

cp infloop.c ../infloop2.c

makes a copy of **infloop.c**, named **infloop2.c**, in the parent directory

Renaming/Moving Files: mv

As before, assume we're in a directory containing a file named **infloop.c**:

mv infloop.c infiniteloop.c

changes the name of the file **infloop.c** to **infinitefloop.c**

mv infloop.c ../attic

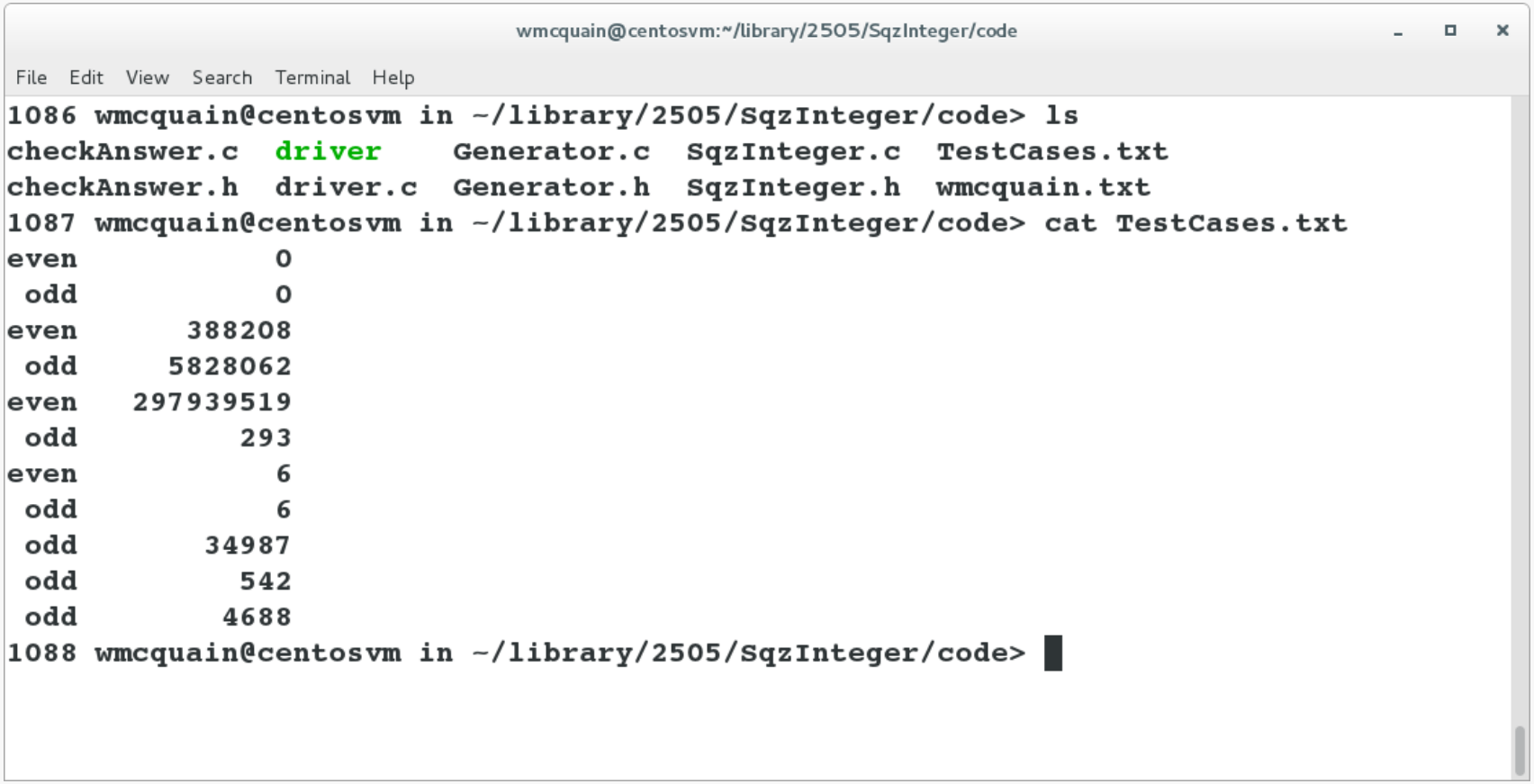
moves the file **infloop.c** to the subdirectory of the parent named **attic**

mv infloop.c ../infiniteloop.c

removes the file **infloop.c** from this directory, and creates a copy named **infinitefloop.c** in the parent directory

Viewing a File: cat

You can use the **cat** command to display the contents of a file to the terminal:



Viewing a File: less

You can use the **less** command to display the contents of a file to the terminal, one screenful at a time; here we entered **less driver.c**:

wmcquain@centosvm:~/library/2505/SqzInteger/code

File Edit View Search Terminal Help

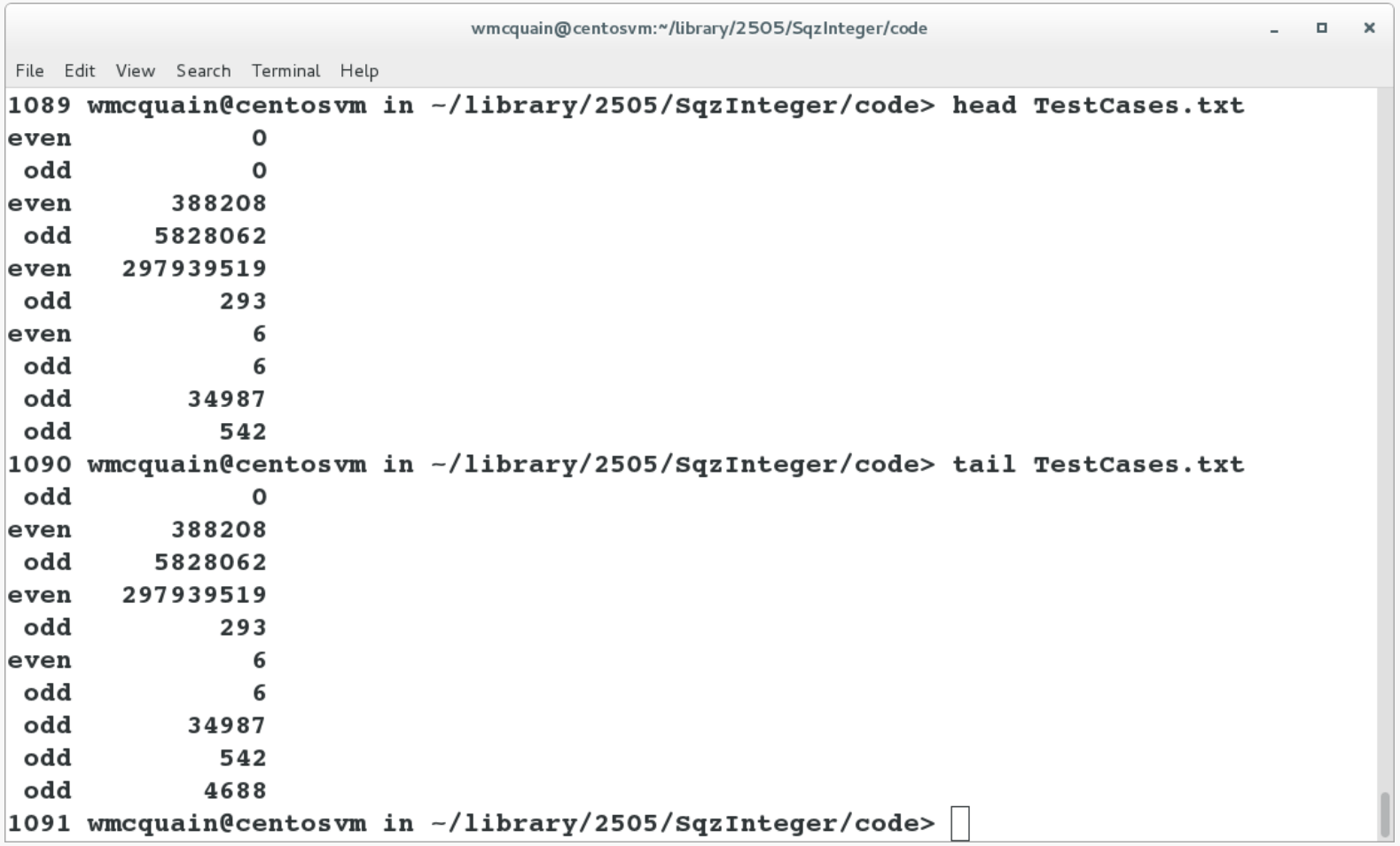
```
// CS 2505 Fall 2017
// Test driver for C01
//
// Invocation:   ./driver <PID> [-repeat]
//
// Standard Library headers:
#include <stdio.h>          // for I/O functionality
#include <stdlib.h>         // for general features
#include <string.h>         // for C string library functionality
// "Local" headers:
#include "SqzInteger.h"     // for assigned function
#include "Generator.h"      // for test case generation
#include "checkAnswer.h"    // for grading functions

// Test driver for the SqzInteger() function.
// Invoke as:
//              driver <PID>
//
// The driver calls a test-case generator, which creates an input file
// containing a fixed number of test cases.  Then, your version of the
// function SqzInteger() is applied to each test case, and the results
// are passed to a check tool.  The check tool uses a reference version
// of the function to compute the correct solution, and compare that to
driver.c
```

Just hit **<space>** or **f** to advance, **b** to back up, and **q** to quit.

Viewing a File: head and tail

You can view the first (or last) few lines of a file by using the **head** (or **tail**) command:



You can control how many lines are shown; see the **man** page.

Counting Words… and More

The **wc** command reports the number of lines, "words", and bytes in a file:

wmcquain@centosvm:~/library/2505/SqzInteger/code

File Edit View Search Terminal Help

```
1023 wmcquain@centosvm in ~/library/2505/SqzInteger/code> wc driver
   21   223 13749 driver
1024 wmcquain@centosvm in ~/library/2505/SqzInteger/code> wc *
   67   222  1493 checkAnswer.c
   20    88   614 checkAnswer.h
   21   223 13749 driver
  161   984  6407 driver.c
  126   424  2677 Generator.c
   14    61   363 Generator.h
   40   168   982 SqzInteger.c
   26   149   851 SqzInteger.h
   11    22   187 TestCases.txt
   21    84   978 wmcquain.txt
  507  2425 28301 total
1025 wmcquain@centosvm in ~/library/2505/SqzInteger/code>
```

Searching File Contents: grep

The **grep** command can be used to display lines of a file that match a pattern:

wmcquain@centosvm:~/2505/C01/code

File Edit View Search Terminal Help

```
1111 wmcquain@centosvm in ~/2505/C01/code> ls -l
total 32
-rw-rw----. 1 wmcquain wmcquain 1493 Aug  2 20:44 checkAnswer.c
-rw-rw----. 1 wmcquain wmcquain  614 Aug  2 20:44 checkAnswer.h
-rw-rw----. 1 wmcquain wmcquain 6407 Aug 18 14:02 driver.c
-rw-rw----. 1 wmcquain wmcquain 2677 Aug  2 20:41 Generator.c
-rw-rw----. 1 wmcquain wmcquain  363 Aug  2 20:41 Generator.h
-rw-rw----. 1 wmcquain wmcquain  982 Aug  2 20:45 SqzInteger.c
-rw-rw----. 1 wmcquain wmcquain  851 Aug  2 20:36 SqzInteger.h
1112 wmcquain@centosvm in ~/2505/C01/code> grep -n return driver.c
47:       return 1;
67:       return 2;
76:       return 3;
89:    // return NULL If it fails to read any input.  That will force the
94:       // read into Line; sscanf() will return 0 if nothing was read.
124:       // strcmp() compares two (C-style) character strings; it returns
159:    // Traditionally, return 0 on successful termination.
160:    return 0;
1113 wmcquain@centosvm in ~/2505/C01/code>
```

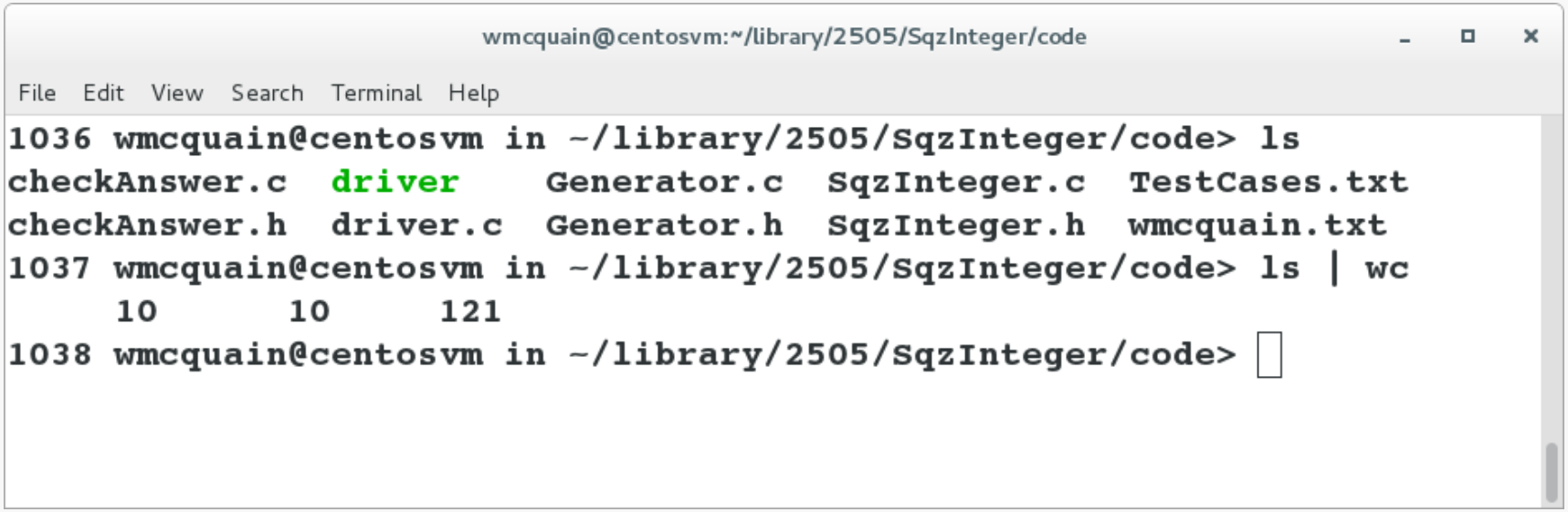
Searching File Contents: grep

The **grep** command can also be used to examine a collection of files:

```
wmcquain@centosvm:~/2505/C01/code
File  Edit  View  Search  Terminal  Help
1114 wmcquain@centosvm in ~/2505/C01/code> ls
checkAnswer.c  driver.c     Generator.h   SqzInteger.h
checkAnswer.h  Generator.c  SqzInteger.c
1115 wmcquain@centosvm in ~/2505/C01/code> grep -n fprintf *.c
checkAnswer.c:20:                fprintf(fp, "Correct!      ");
checkAnswer.c:24:                fprintf(fp, "Alas, wrong:  ");
checkAnswer.c:26:        fprintf(fp, "%12"PRIu32, origValue);
checkAnswer.c:28:                fprintf(fp, "   even  ");
checkAnswer.c:31:                fprintf(fp, "    odd  ");
checkAnswer.c:33:        fprintf(fp, "%12"PRIu32, stuAnswer);
checkAnswer.c:36:                fprintf(fp, "  should be %"PRIu32, refAnswer);
checkAnswer.c:38:        fprintf(fp, "\n");
driver.c:136:          fprintf(results, "Unrecognized action flag: %s\n", actionFlag)
;
driver.c:149:    fprintf(results, "Score:  %3"PRIu32" / %3"PRIu32"\n", totalScore, to
talPts);
Generator.c:125:    fprintf(Out, "%4s  %10d\n", Flag, N);
1116 wmcquain@centosvm in ~/2505/C01/code>
```

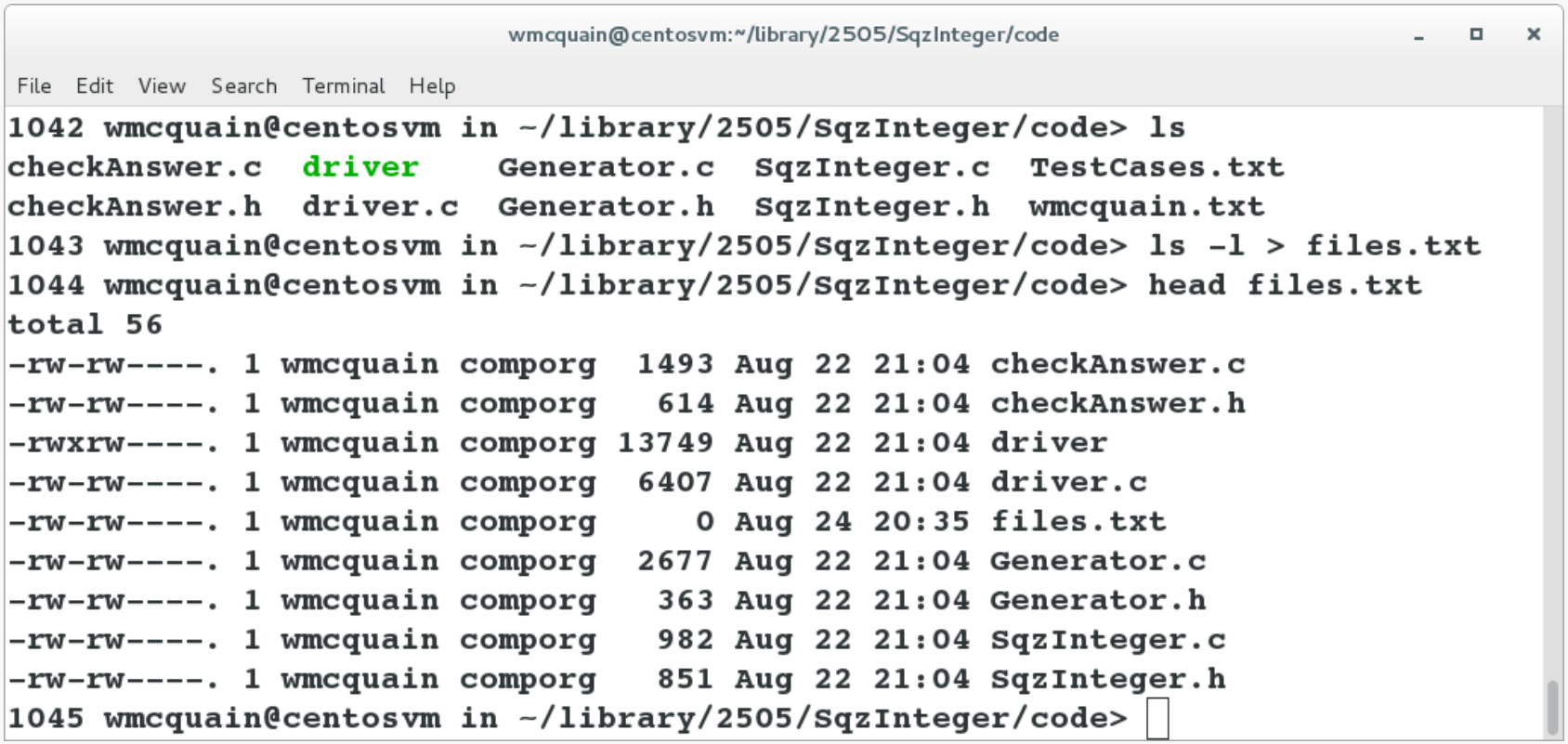
Chaining Commands

The **pipe** symbol (**|**) connects standard output from one command to standard input for the next command:



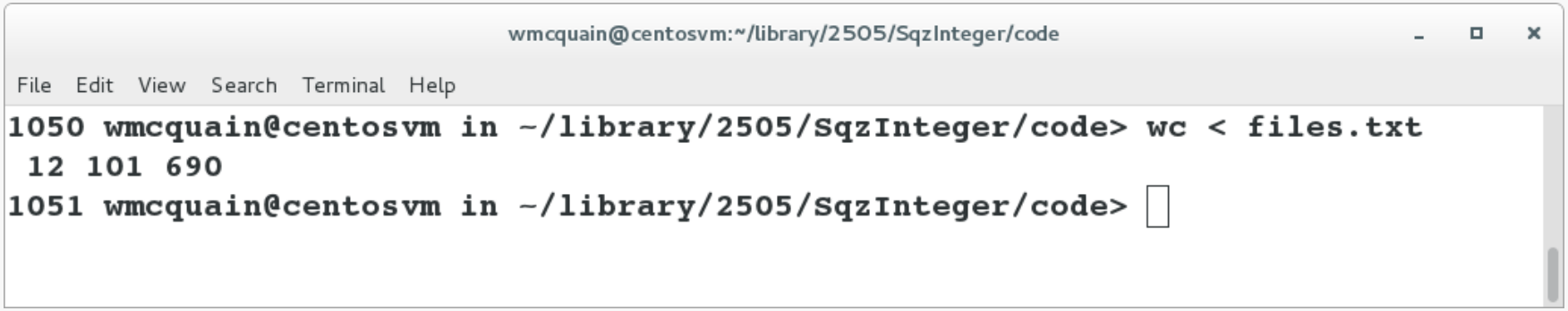
Redirecting Output

The output a program writes to standard output (the terminal) can be sent to a file by using an **output redirection operator** (**>** or **>>**):



Redirecting Input

The contents of a file can be sent, as standard input, to a program by using an **input redirection operator** (**<** or **<<**):



Checking the File Type: file

You can obtain information about a file with the **file** command:

wmcquain@centosvm:~/2505/C01/code

File Edit View Search Terminal Help

```
1123 wmcquain@centosvm in ~/2505/C01/code> ls
checkAnswer.c  driver    Generator.c  SqzInteger.c  TestCases.txt
checkAnswer.h  driver.c  Generator.h  SqzInteger.h  wmcquain.txt
1124 wmcquain@centosvm in ~/2505/C01/code> file TestCases.txt
TestCases.txt: ASCII text
1125 wmcquain@centosvm in ~/2505/C01/code> file driver
driver: ELF 64-bit LSB executable, x86-64, version 1 (SYSV), dynamically linked (use
s shared libs), for GNU/Linux 2.6.32, BuildID[sha1]=dea893278362719fb481691010cd5632
bebf18b3, not stripped
1126 wmcquain@centosvm in ~/2505/C01/code>
```

Traditional Access Permissions

There are three types of users:

- *owner (aka user)*
- *group*
- *other (aka world)*

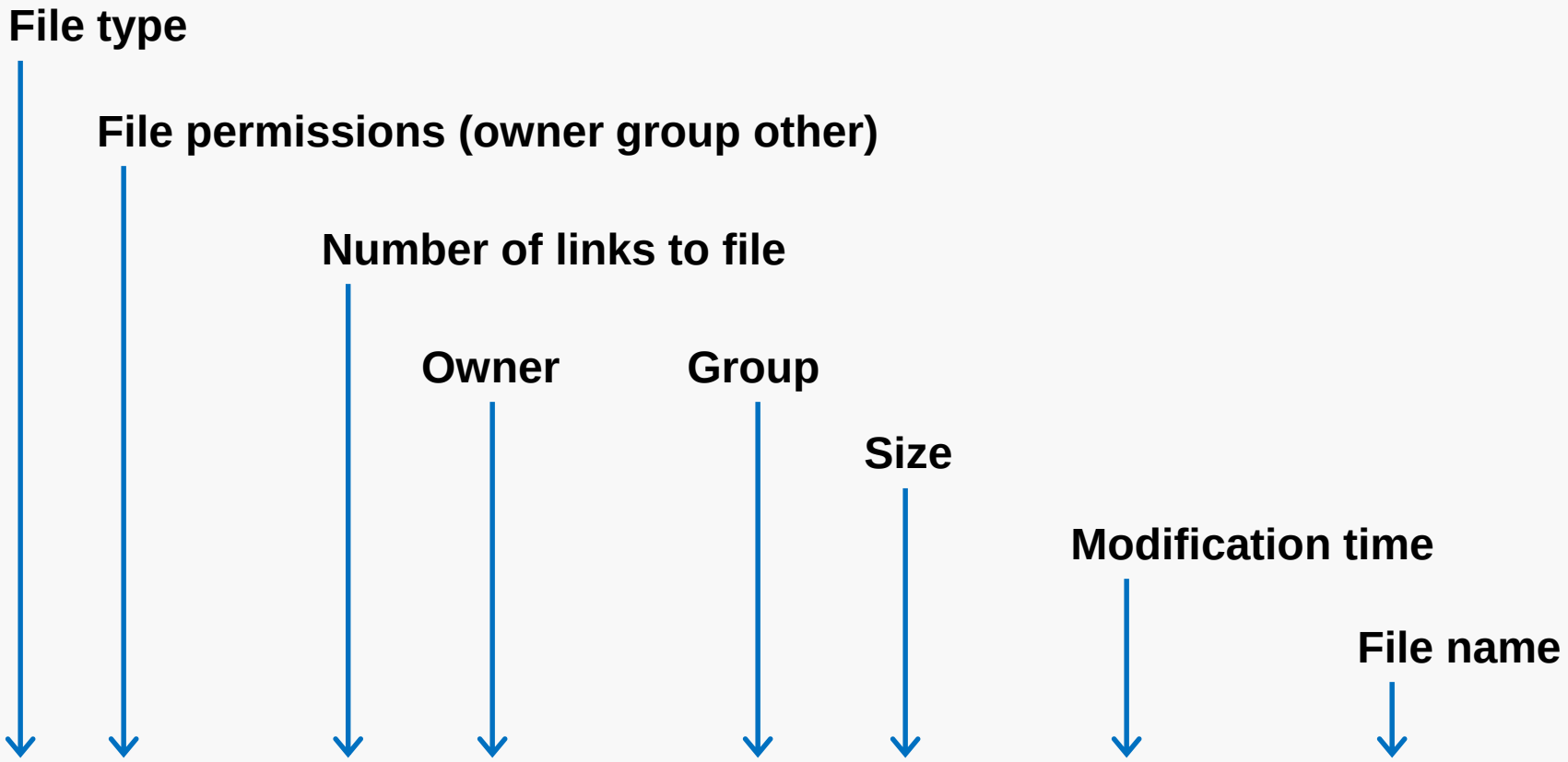
A user may attempt to access an ordinary file in three ways:

- *read from*
- *write to*
- *execute*

Use **ls –l** to view the file permissions:

```
1143 wmcquain@centosvm in ~/2505> ls -l C04/code
total 48
-rw-rw----. 1 wmcquain comporg  4653 Aug  7 21:45 comparator.c
-rwxr-xr-x. 1 wmcquain comporg 13340 Aug  7 21:45 compare
-rw-rw----. 1 wmcquain comporg  2612 Aug  7 21:45 driver.c
-rw-rw----. 1 wmcquain comporg   451 Aug 10 19:30 Intersection.h
-rw-rw----. 1 wmcquain comporg  1944 Aug  7 21:45 Intersection.o
drwxrwxr--. 1 wmcquain comporg  3073 Aug  7 21:45 tmpbackup
```

Traditional Access Permissions




```
-rwxr-xr-x. 1 wmcquain comporg 13340 Aug  7 21:45 compare
drwxrwxr--. 1 wmcquain comporg  3073 Aug  7 21:45 tmpbackup
```

Changing Access Permissions: chmod

Use the **chmod** command to set or alter traditional file permissions:

```
1143 wmcquain@centosvm in ~/2505> ls -l C04/code/comparator.c
-rw-rw----. 1 wmcquain comporg  4653 Aug  7 21:45 comparator.c

1144 wmcquain@centosvs in ~/2505> chmod g-rw C04/code/comparator.c

1143 wmcquain@centosvm in ~/2505> ls -l C04/code/comparator.c
-rw-------. 1 wmcquain comporg  4653 Aug  7 21:45 comparator.c
```

chmod also allows the use of numeric arguments:

- **0** no access permissions
- **1** execute permissions
- **2** write to permissions
- **4** read from permissions

So, **chmod 740** would set

owner permissions to **r w x**
group permissions to **r- -**
other permissions to **- - -**

WHY?

Changing Access Permissions: chmod

Binary representations:

none	**0**	**000**
x	**1**	**001**
w	**2**	**010**
r	**4**	**100**

Now notice that **7 = 111** which is the logical OR of **001** and **010** and **100**

And, **740** thus specifies permissions **7** for the owner, **4** for the group and **0** for others.

The Importance of Access Permissions

When working on a shared environment, like the rlogin cluster, it is vital that you make sure that your access permissions are set correctly.

As a general rule, you will rely on the default access permissions, which are controlled via shell configuration files we will discuss later.

When in doubt, use **ls –l** to check!

Bundling Files into an Archive: tar

You can create a single file that contains a collection of files, including a directory structure with the **tar** utility:

```
wmcquain@centosvm:~/2505/C01/code
File  Edit  View  Search  Terminal  Help
1130 wmcquain@centosvm in ~/2505/C01/code> ls
checkAnswer.c  driver.c     Generator.h   SqzInteger.h
checkAnswer.h  Generator.c  SqzInteger.c
1131 wmcquain@centosvm in ~/2505/C01/code> tar cvf C01Code.tar *.c *.h
checkAnswer.c
driver.c
Generator.c
SqzInteger.c
checkAnswer.h
Generator.h
SqzInteger.h
1132 wmcquain@centosvm in ~/2505/C01/code> ls
C01Code.tar    checkAnswer.h  Generator.c  SqzInteger.c
checkAnswer.c  driver.c       Generator.h  SqzInteger.h
1133 wmcquain@centosvm in ~/2505/C01/code>
```

cvf **create archive, be *v*erbose, write to a file**

Note the name of the new tar file is listed before the target (files to be tar'd up).

DO NOT get that backwards!

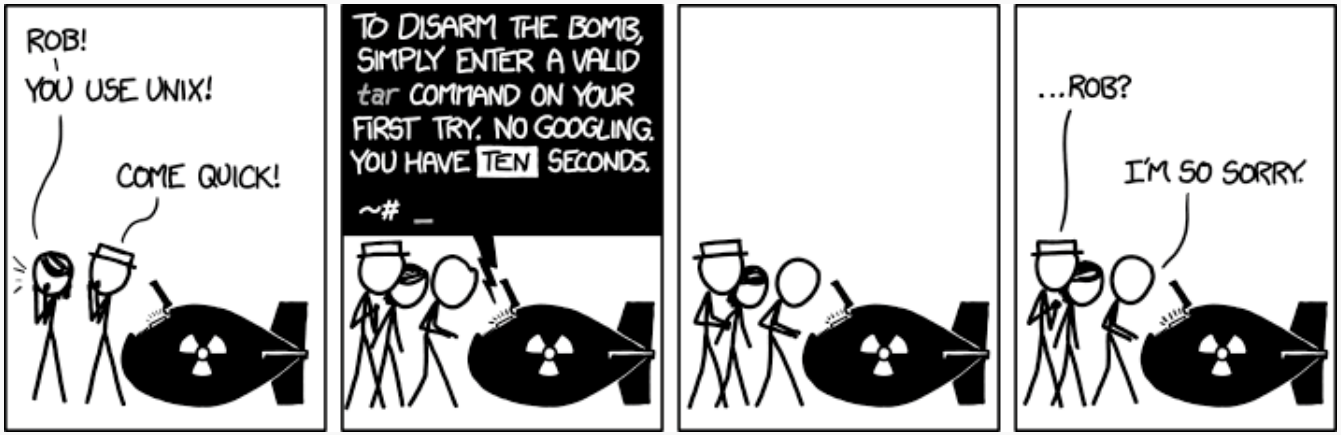
Applied Terror: tar

As with all commands, your syntax must be precise… but the tar command has the potential to destroy files:

wmcquain@centosvm:~/2505/C01/code/temp

File Edit View Search Terminal Help

```
1147 wmcquain@centosvm in ~/2505/C01/code/temp> ls
C01Code.tar  driver.c  SqzInteger.c  SqzInteger.h
1148 wmcquain@centosvm in ~/2505/C01/code/temp> tar cvf driver.c SqzInteger.c SqzInt
eger.h C01Code.tar
SqzInteger.c
SqzInteger.h
C01Code.tar
1149 wmcquain@centosvm in ~/2505/C01/code/temp> ls
C01Code.tar  driver.c  SqzInteger.c  SqzInteger.h
1150 wmcquain@centosvm in ~/2505/C01/code/temp> file driver.c
driver.c: POSIX tar archive (GNU)
1151 wmcquain@centosvm in ~/2505/C01/code/temp>
```



xkcd.org

A Safer Way

There is a bash shell script on the Resources page that provides a safer alternative:

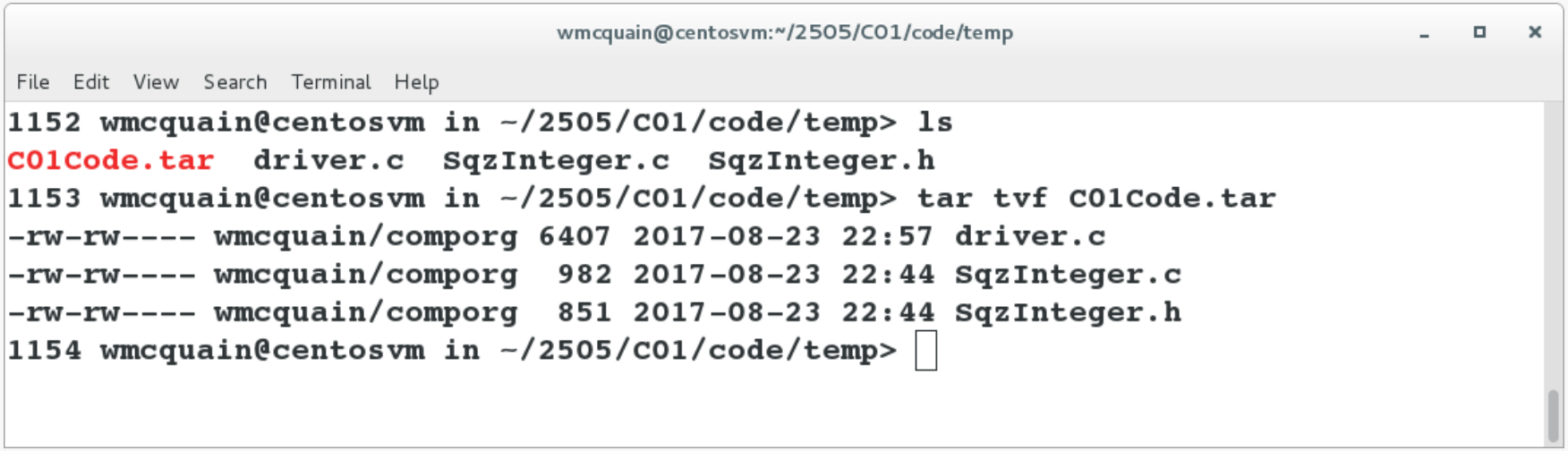
```
wmcquain@centosvm:~/2505/C01/code/temp
File  Edit  View  Search  Terminal  Help
1168 wmcquain@centosvm in ~/2505/C01/code/temp> ls
C01Code.tar  driver.c  SqzInteger.c  SqzInteger.h
1169 wmcquain@centosvm in ~/2505/C01/code/temp> safertar.sh driver.c SqzInteger.c Sq
zInteger.h C01Code.tar
driver.c already exists
driver.c IS NOT a tar file
Do you want to overwrite (destroy) driver.c?  Y/N
N
You said: N
1170 wmcquain@centosvm in ~/2505/C01/code/temp>
```

Download `safertar.sh`, put it in a directory in your path, and make it executable.

This comes with the usual software license…

Checking Contents

You can check the contents of a tar file:

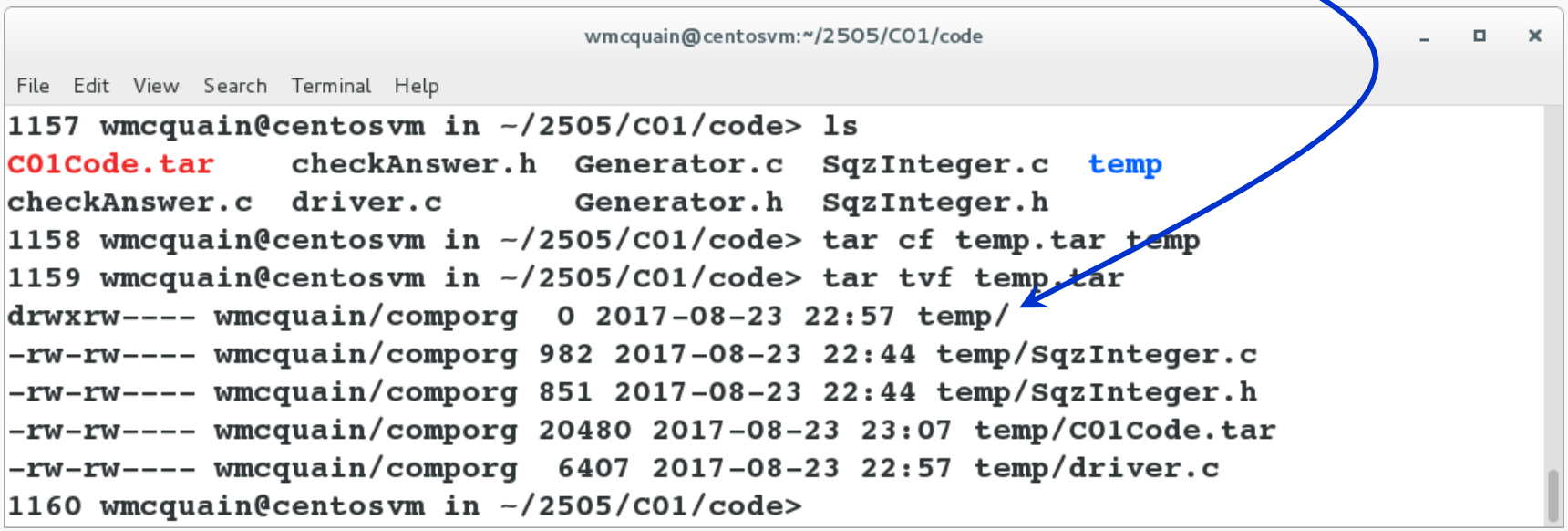


This is an example of a *flat* tar file.

That is, there is no directory structure in the tar file.

A tar File That Is Not Flat

If you tar a directory tree, the tar file will (by default) contain directory information:



Some situations require a flat tar file, some require creating one that preserves a directory structure.

Be sure you pay attention to what's required, and create the correct type of tar.

Extracting a tar File

Use the **x** switch to extract the contents of a **tar** file, and **-C** to specify a destination:

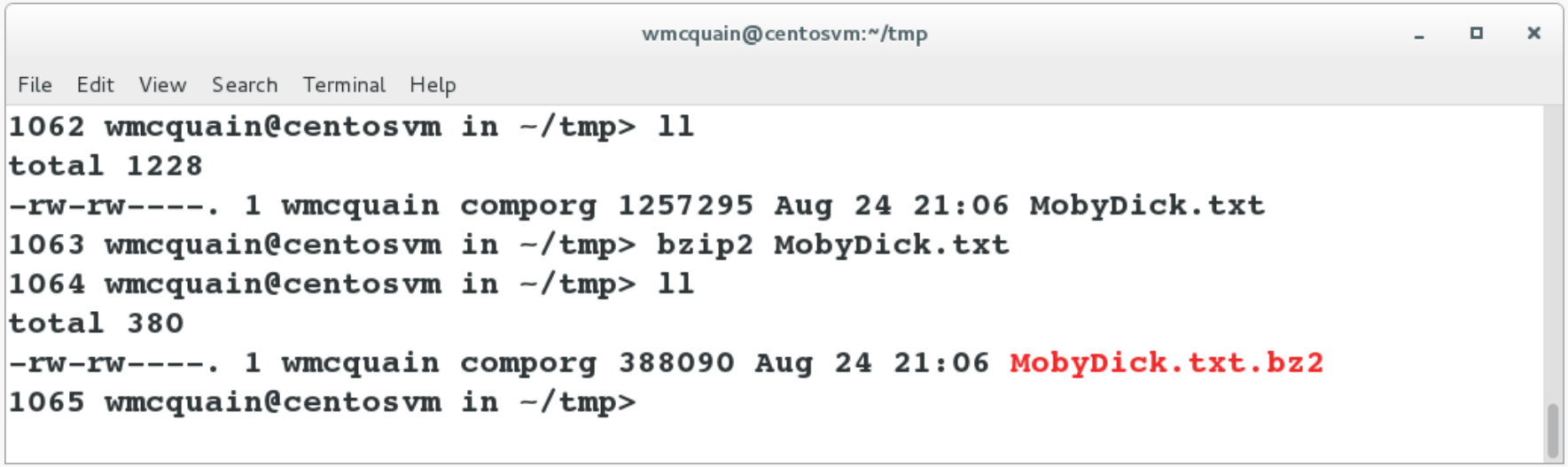
```
wmcquain@centosvm:~/2505/C01/code
File  Edit  View  Search  Terminal  Help
1161 wmcquain@centosvm in ~/2505/C01/code> mkdir xtemp
1162 wmcquain@centosvm in ~/2505/C01/code> ls
C01Code.tar     checkAnswer.h  Generator.c  SqzInteger.c  temp       xtemp
checkAnswer.c   driver.c       Generator.h  SqzInteger.h  temp.tar
1163 wmcquain@centosvm in ~/2505/C01/code> tar xvf temp.tar -C xtemp
temp/
temp/SqzInteger.c
temp/SqzInteger.h
temp/C01Code.tar
temp/driver.c
1164 wmcquain@centosvm in ~/2505/C01/code> ls xtemp
temp
1165 wmcquain@centosvm in ~/2505
```

```
wmcquain@centosvm:~/2505/C01/code
File  Edit  View  Search  Terminal  Help
1167 wmcquain@centosvm in ~/2505/C01/code> tree xtemp
xtemp
`-- temp
    |-- C01Code.tar
    |-- driver.c
    |-- SqzInteger.c
    `-- SqzInteger.h

1 directory, 4 files
1168 wmcquain@centosvm in ~/2505/C01/code>
```

Compressing Files: gzip, bzip2, etc.

A compression tool can frequently reduce the amount of space a file occupies:



A common, older alternative is **gzip**.

Both of these suffer the same limitation: they can compress, but not bundle.

Therefore, it's common to create a tar archive and then compress that.

Compressing Files: zip

The **zip** utility also compresses, but will bundle as well:

wmcquain@centosvm:~/2505/C01/code

File Edit View Search Terminal Help

```
1074 wmcquain@centosvm in ~/2505/C01/code> ll
total 32
-rw-rw----. 1 wmcquain wmcquain 1493 Aug  2 20:44 checkAnswer.c
-rw-rw----. 1 wmcquain wmcquain  614 Aug  2 20:44 checkAnswer.h
-rw-rw----. 1 wmcquain wmcquain 6407 Aug 18 14:02 driver.c
-rw-rw----. 1 wmcquain wmcquain 2677 Aug  2 20:41 Generator.c
-rw-rw----. 1 wmcquain wmcquain  363 Aug  2 20:41 Generator.h
-rw-rw----. 1 wmcquain wmcquain  982 Aug  2 20:45 SqzInteger.c
-rw-rw----. 1 wmcquain wmcquain  851 Aug  2 20:36 SqzInteger.h
1075 wmcquain@centosvm in ~/2505/C01/code> zip C01Code.zip *.c *.h
  adding: checkAnswer.c (deflated 60%)
  adding: driver.c (deflated 62%)
  adding: Generator.c (deflated 72%)
  adding: SqzInteger.c (deflated 52%)
  adding: checkAnswer.h (deflated 43%)
  adding: Generator.h (deflated 32%)
  adding: SqzInteger.h (deflated 42%)
1076 wmcquain@centosvm in ~/2505/C01/code>
```

Compressing Files: zip

The degree of compression depends on the nature of the file(s) being compressed:

wmcquain@centosvm:~/2505/C01/code

File Edit View Search Terminal Help

```
1084 wmcquain@centosvm in ~/2505/C01/code> ls -l C01Code.zip
-rw-rw----. 1 wmcquain comporg 6390 Aug 24 21:23 C01Code.zip
1085 wmcquain@centosvm in ~/2505/C01/code> unzip -l C01Code.zip
Archive:  C01Code.zip
  Length      Date    Time    Name
---------  ---------- -----   ----
     1493  08-02-2017 20:44   checkAnswer.c
     6407  08-18-2017 14:02   driver.c
     2677  08-02-2017 20:41   Generator.c
      982  08-02-2017 20:45   SqzInteger.c
      614  08-02-2017 20:44   checkAnswer.h
      363  08-02-2017 20:41   Generator.h
      851  08-02-2017 20:36   SqzInteger.h
---------                     -------
    13387                     7 files
1086 wmcquain@centosvm in ~/2505/C01/code>
```

Uncompressing: gunzip, bunzip2, unzip

Each compression tool has an analog that will uncompress; for example:

```
wmcquain@centosvm:~/2505/C01/code
File  Edit  View  Search  Terminal  Help
1093 wmcquain@centosvm in ~/2505/C01/code> ls
C01Code.zip    checkAnswer.h  Generator.c  SqzInteger.c  unpacked
checkAnswer.c  driver.c       Generator.h  SqzInteger.h
1094 wmcquain@centosvm in ~/2505/C01/code> unzip C01Code.zip -d unpacked
Archive:  C01Code.zip
  inflating: unpacked/checkAnswer.c
  inflating: unpacked/driver.c
  inflating: unpacked/Generator.c
  inflating: unpacked/SqzInteger.c
  inflating: unpacked/checkAnswer.h
  inflating: unpacked/Generator.h
  inflating: unpacked/SqzInteger.h
1095 wmcquain@centosvm in ~/2505/C01/code> ls unpacked/
checkAnswer.c  driver.c     Generator.h   SqzInteger.h
checkAnswer.h  Generator.c  SqzInteger.c
1096 wmcquain@centosvm in ~/2505/C01/code>
```

Removing a File: shred and dd

rm does not actually delete file contents from your drive; it just deindexes the file.

You can securely remove a file by using the **shred** command, but see Sobell for a discussion of the limitations.

See the discussion of **dd** in Sobell for an alternative way to wipe a file.

Special Characters

Many Linux commands support the use of special characters (aka wildcards) to specify a pattern that identifies a set of files:

?	matches any single character (in the name of an existing file)
*****	matches zero or more characters (in the name of an existing file)
[ ]	matches any of the characters within the braces (in the name of an existing file)

***.txt**

matches any file with extension "txt"

foo?.*

matches a file with any extension and name consisting of "foo" followed by a single character

[abc]foo.html

matches a file with extension "html" and name "afoo" or "bfoo" or "cfoo"

Copying a File Remotely: scp

scp can be used to copy a file between the local machine and a remote machine (or between two remote machines).

For example, the following command would copy `GettysburgAddress.txt` from my computer to a directory named **documents** on **rlogin**:

scp GettysburgAddress.txt wmcquain@rlogin.cs.vt.edu:documents

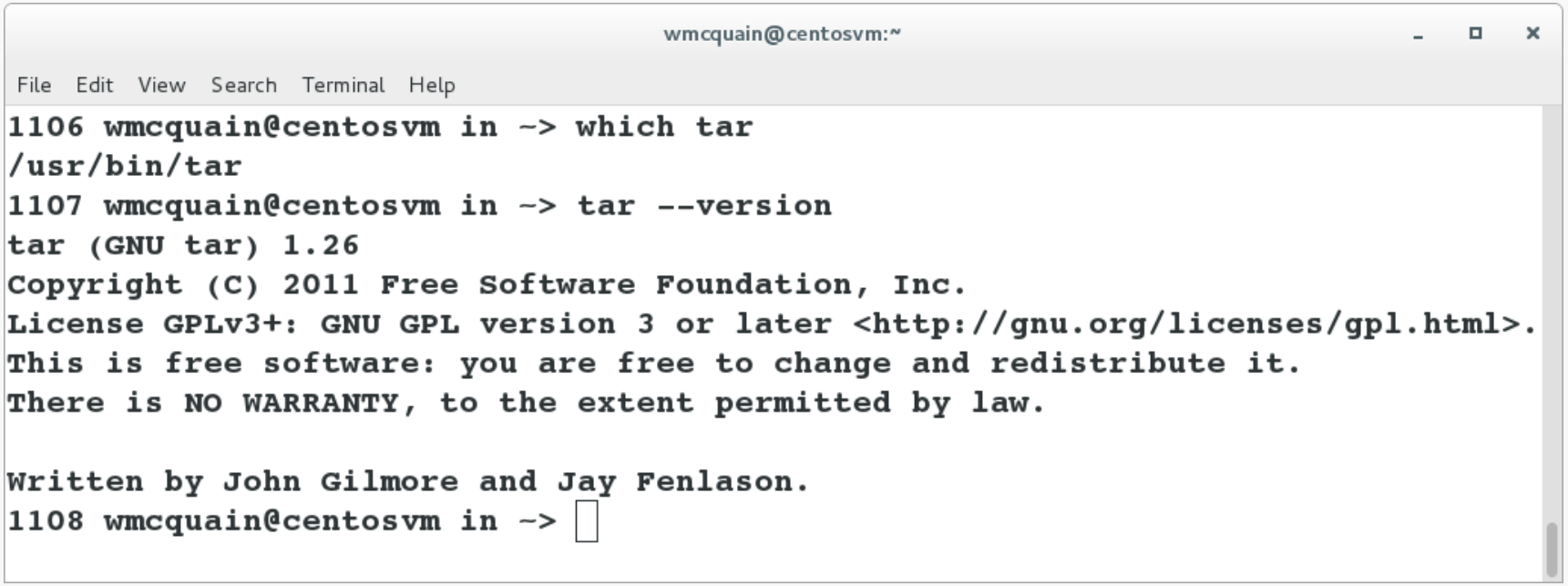
If you haven't set up password-less login, you'll be prompted for the necessary authentication information.

And the following command would copy `GettysburgAddress.txt` from my rlogin account to my current directory on my machine:

scp wmcquain@rlogin.cs.vt.edu:documents/GettysburgAddress.txt .

Identifying a Command: which

If you're not sure where a command resides, the **which** command will tell you:



Many Linux applications also support a **--version** switch which can help identify which specific version of an application you're invoking.

Foreground vs Background

By default when you execute a command in a shell, the shell program waits (doesn't provide a prompt and allow entry of another command) until the current command completes (or is otherwise interrupted).

Here, the command is running in the *foreground*:

```
1141 wmcquain@centosvm in ~/tmp> sleeper 5 hi
  0   995:  hi
  1   997:  hi
  2     1:  hi
  3     7:  hi
  4    15:  hi
```

Foreground vs Background

We may run the command in the *background*:

```
1143 wmcquain@centosvm in ~/tmp> sleeper 5 hi > sleeper.txt &
[1] 8672

1144 wmcquain@centosvm in ~/tmp> ps
  PID TTY          TIME CMD
 3928 pts/0    00:00:01 bash
 8672 pts/0    00:00:00 sleeper
 8676 pts/0    00:00:00 ps

1145 wmcquain@centosvm in ~/tmp>
[1]+  Done                    sleeper 5 hi > sleeper.txt

1145 wmcquain@centosvm in ~/tmp> cat sleeper.txt
  0   270:  hi
  1   272:  hi
  2   276:  hi
  3   282:  hi
  4   290:  hi
```

Killing a Process

A (foreground) running process can be killed by using **Ctrl-C**.

A (background) running process or a suspended process can be killed by using the **kill** command: