Files and Directories

Unit objectives

After completing this unit, you should be able to:

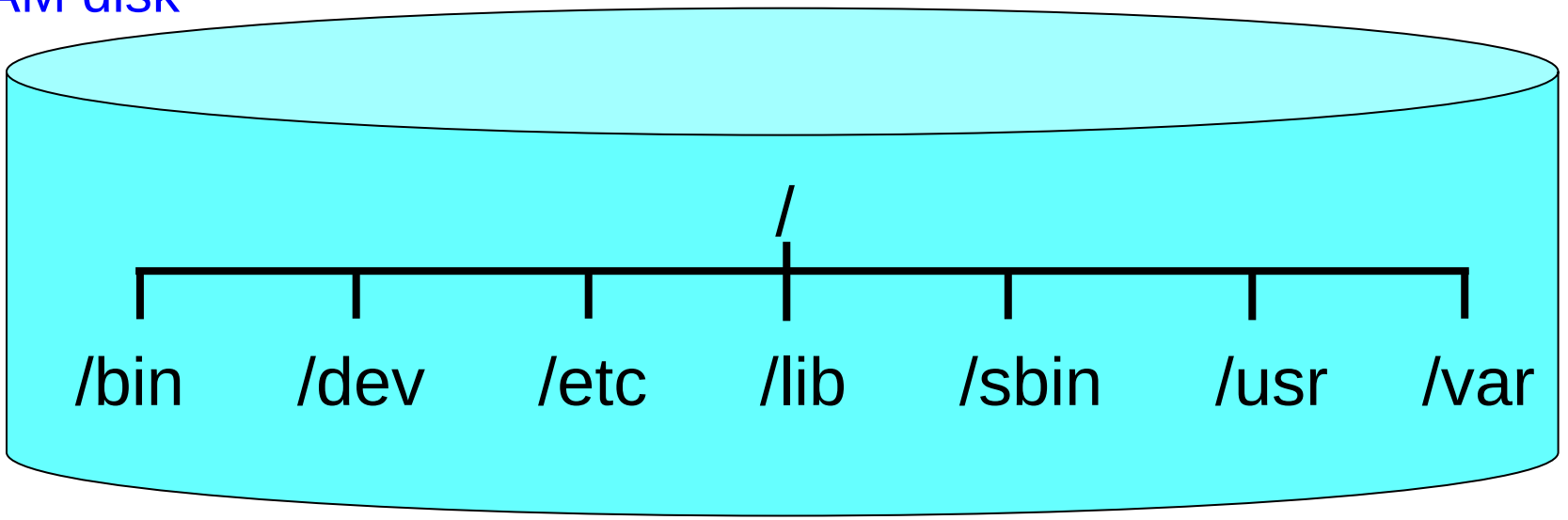
- Describe what a file is
- Describe what a file system is
- List possible file systems
- Describe i-nodes
- Create/mount/unmount file systems
- Create predefined mounts
- Set up user and group quota

What is a file?

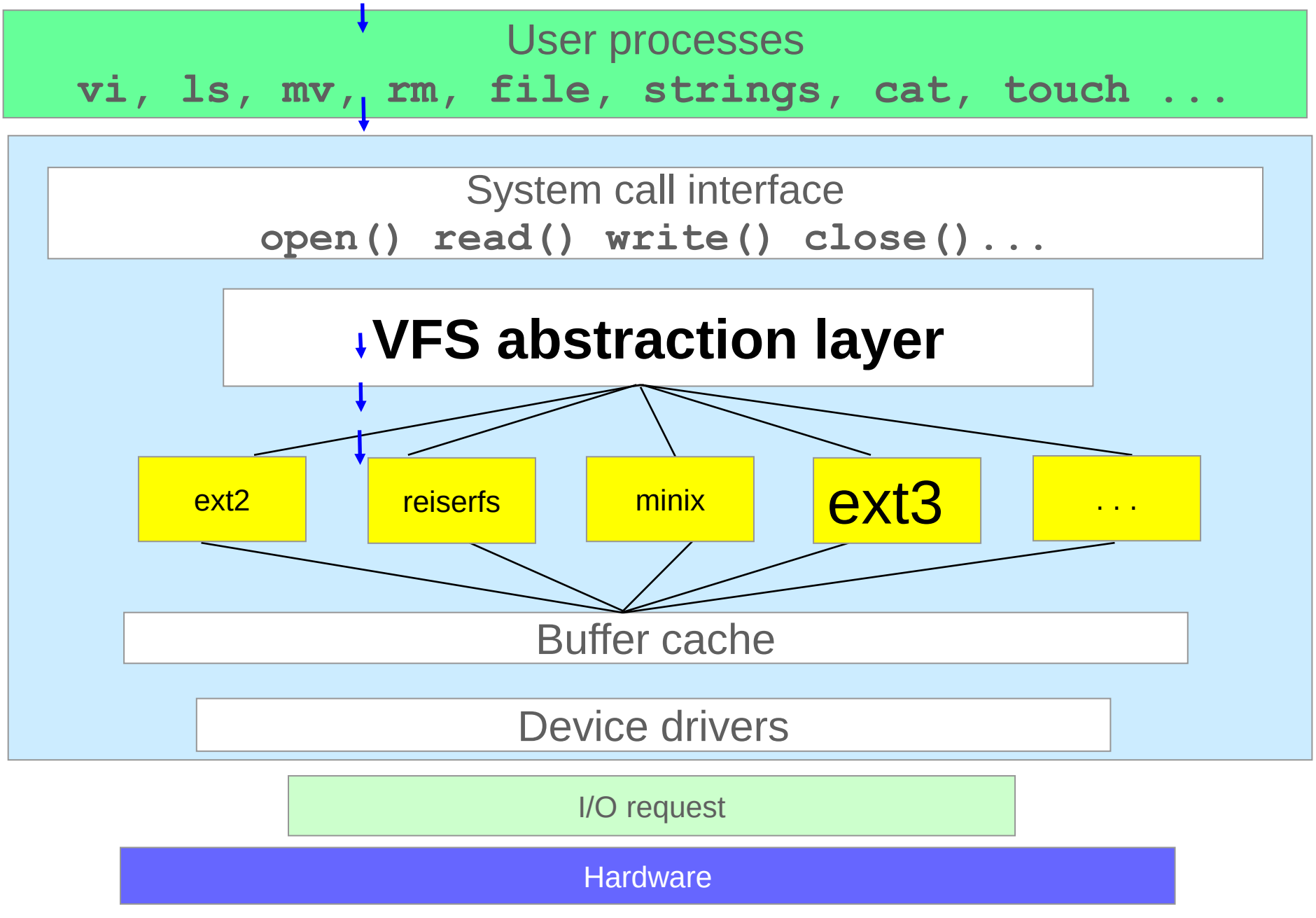
- Consecutive number of bytes
  - No internal structure by default (applications define their own structure)
- Stored and referenced in a file system
  - Can have multiple references (names)
- Special files exist
  - Block, Character > Device
  - Pipes, Sockets > Interprocess communication

What is a file system?

- Place to store files and refer to them
- Hierarchical structure through use of directories
- A file system can be stored on any block device
  - Floppy disk
  - Hard disk
  - Partition
  - RAID, LVM volume
  - File (for use with a loop device)
  - RAM disk



The virtual file system



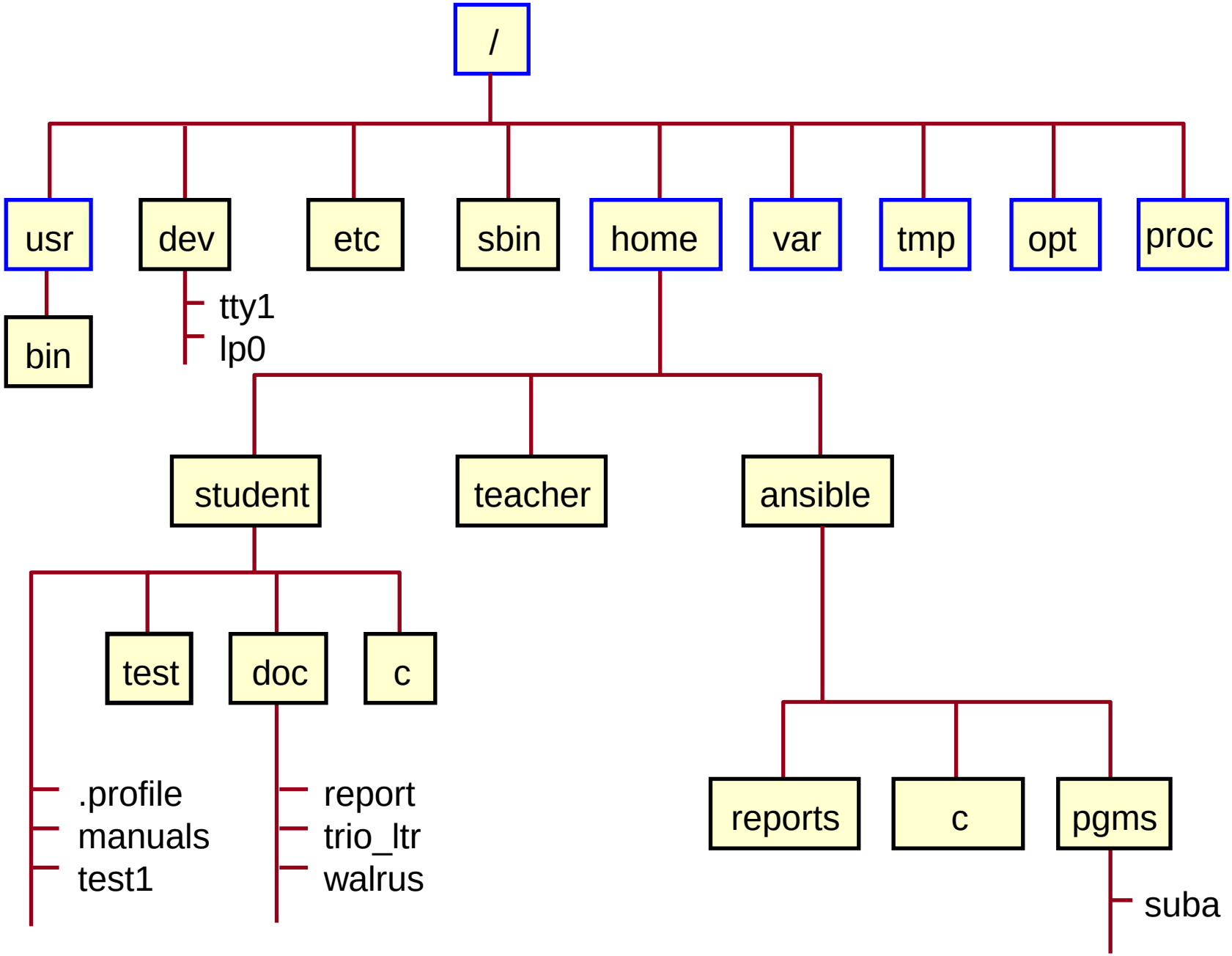
File systems supported

- Traditional: ext2
- Second generation: ext3, ReiserFS, IBM JFS, xfs
- Next generation: ext4, GFS2, Reiser4
- FAT-12, FAT-16, FAT-32, VFAT, NTFS (read-only)
- CD-ROM (ISO 9660)
- UMSDOS (UNIX-like FS on MS-DOS)
- NFS (Network File System)
- SMBFS (Windows share), NCPFS (Novell NetWare share)
- `/proc` (for kernel and process information)
- SHMFS (shared memory file system)
- GPFS, Lustre (clustering file systems)

Linux file systems

- In Linux, a *file system* is an *allocation of storage*
- Refers to both the physical and logical storage and access of files
- Similar in concept to partitions in the PC environment
- Allows the operating system to store and retrieve the data from files quickly and efficiently
- To access file systems, you associate them with a directory
- Linux has several pre-defined file systems
  - `/(root)`
  - `/usr`
  - `/var`
  - `/tmp`
  - `/home`
  - `/admin`
  - `/proc`
  - `opt`

Hierarchical structure



Path names

- A sequence of names, which are separated by slashes (/) that describes the path that the system must follow to locate a file in the file system
- There are two types of path names:
  - Absolute or full path name (start from the / directory):

```
$ vi /home/student/doc/report
$ /usr/bin/ls -l /home/student
```

  - Relative path name (start from current directory):

```
$ cd /home/student
$ vi doc/report
$ cd /usr/bin
$ ./ls -l /home/student
```

Where am I?

- The `pwd` (print working directory) command can be used to find out what your current directory is.

```
$ pwd
/home/student
```

Listing directories

- The `ls` command displays the contents of a directory
- Basic syntax: `ls [`*`directory`*`]`
- Common options:
  - `-a` Show hidden files (files that start with a ".")
  - `-R` List files in all subdirectories (recursively)

```
$ ls
c doc manuals test1

$ ls -a
. .. .profile c doc manuals test1

$ ls -R
c doc manuals test1

./c:
./doc:
report trio_ltr walrus
```

Change current directory

- The `cd` command changes the current directory
- Basic syntax: `cd [directory]`
- Set the current working directory from `/home/student` to `/home/student/doc`:
  - Using a relative path:

```
$ cd doc
```

  - Using a full (absolute) path:

```
$ cd /home/student/doc
```

- Set your working directory to your home directory:

```
$ cd
```

- Set your working directory to the parent directory:

```
$ cd ..
```

Long listing of files

- The `ls` command with the `-l` option can be used to obtain more information about the files in a directory.

```
$ ls -l
total 40
drwxr-xr-x    2 student   staff           256 Mar 13 13:01 c
drwxr-xr-x    2 student   staff           256 Mar 13 13:13 doc
-rw-r--r--    1 student   staff         13886 Mar 13 13:02 manuals
-rw-r--r--    1 student   staff          3331 Mar 13 13:02 test1

$ ls -li test1
 8208 -rw-r--r--    1 student   staff        3331 Mar 13 13:23
test1
```

Creating directories

- The `mkdir` command creates a new directory
- Basic syntax: `mkdir` *`directory`*
- To create the directory `mydir`, as a subdirectory of `/home/student`:
  - Using a relative path:

```
$ cd /home/student
$ mkdir mydir
```

  - Using a full (absolute) path:

```
$ mkdir /home/student/mydir
```

Removing directories

- The `rmdir` command removes a directory
- Basic syntax: `rmdir` *`directory`*
- To remove the directory `/home/student/mydir`:
  - Using a relative path:

```
$ cd /home/student
$ rmdir mydir
```

  - Using a full (absolute) path:

```
$ rmdir /home/student/mydir
```

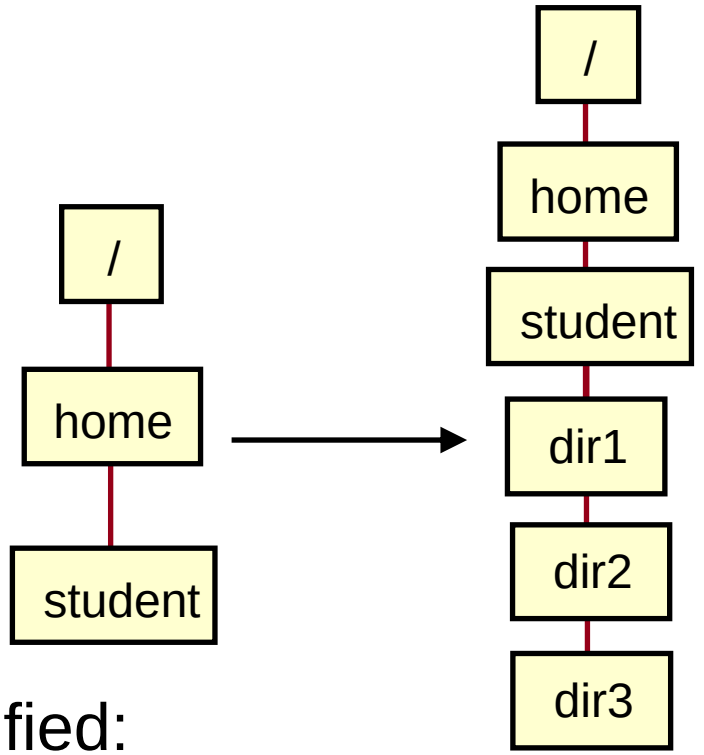
- The directory must be empty!

```
$ rmdir doc
rmdir: Directory doc is not empty.
```

Working with multiple directories

- Create multiple directories simultaneously
  - The `mkdir -p` option creates missing intermediate path name directories

```
$ cd /home/student
$ mkdir -p dir1/dir2/dir3
```



- Remove all directories in the path specified:
  - The `rmdir -p` option removes all directories along the path name specified

```
$ cd /home/student
$ rmdir -p dir1/dir2/dir3
```

Displaying directory information

```
$ ls -ldi doc
 8206 drwxr-xr-x    2 student   staff           256 Mar 13 13:13
doc


$ stat doc
Inode 8206 on device 10/8       Directory
Protection: rwxr-xr-x
Owner: 208(student)              Group: 1(staff)
Link count:   2          Length 256 bytes

Last updated:   Fri Mar 13 13:13:32 2015
Last modified:  Fri Mar 13 13:13:32 2015
Last accessed:  Fri Mar 13 13:18:06 2015
```

Linux file names

- Should be descriptive of the content
- Should use only alphanumeric characters:
  - UPPERCASE, lowercase, number, #, @, _
- Should not include embedded blanks
- Should not contain shell metacharacters:

  `*  ?  >  <  /  ;  &  !  [  ]  |  $  \  '  "  (  )`
- Should not begin with + or - sign
- Should not be the same as a system command
- Are case-sensitive
- File names that start with a . (dot) are hidden from the normal `ls` command
- The maximum number of characters for a file name is 255

The `touch` command

- The `touch` command updates the *access* and *modification times* of a file.
- The command can also be used to *create zero-length* files.

```
$ ls -l
total 40
drwxr-xr-x    2 student   staff           256 Mar 13 13:01 c
drwxr-xr-x    2 student   staff           256 Mar 13 13:13 doc
-rw-r--r--    1 student   staff         13886 Mar 13 13:02 manuals
-rw-r--r--    1 student   staff          3331 Mar 13 13:23 test1

$ date
Mon Mar 13 13:30:06 PDT 2015

$ touch test1 test2

$ ls -l
total 40
drwxr-xr-x    2 student   staff           256 Mar 13 13:01 c
drwxr-xr-x    2 student   staff           256 Mar 13 13:13 doc
-rw-r--r--    1 student   staff         13886 Mar 13 13:02 manuals
-rw-r--r--    1 student   staff          3331 Mar 13 13:30 test1
-rw-r--r--    1 student   staff             0 Mar 13 13:30 test2
```

Unit summary

Having completed this unit, you should understand:

- Describe what a file is
- Describe what a file system is
- List possible file systems
- Create, list and delete files
- Create, change and delete directories