CS241 #29 – Files, Directories, symlinks #3

What do the following do?

```
chmod 600 fileA
chown angrave fileB
chown -R angrave .
chmod o-rwx fileC      # Hint : u=user,g=group,o=other
```

How do I find out if an inode is a regular file or directory or something else?

Problem: How do I recurse into subdirectories? (+ Fix any errors)

```
void dirlist(char*path) {

  struct dirent* dp;
  DIR* dirp = opendir(path);

  while ((dp = readdir(dirp)) != NULL) {

    char newpath[strlen(path)+strlen(dp->d_name)+1];

     sprintf(newpath,"%s/%s", newpath, dp->d_name);

     printf("%s%s \n", dp->d_name);

     dirlist(newpath);
  }
}
int main(int argc,
char**argv){dirlist(argv[1]);return 0;}
```

Fixes required / Notes:

> Symbolic links?

How do they work?

How do I make one?

How do I use readlink?

Why use lstat() instead of stat() ?

> Symbolic vs Hard links Gameshow

advantages? disadvantages?

> Why would I want to set a directory's sticky bit?

> How do I set the sticky bit?

> Which directory will have the sticky bit set?

> What does ‘env’ do?

> **Why do shell programs start with**
`#!/usr/bin/env python`

> How do I make 'hidden' files i.e. not listed by "ls"? How do I list them?

> File permissions and directories

>File system mounts and virtual file systems

> Copying byte streams with dd

```
dd if=/dev/urandom of=~/secret.txt  bs=1k count=1024
dd if=/dev/zero of=~/secret.txt  bs=1k count=1024
dd if=/dev/zero of=/dev/null bs=1m count=1024
```

Examples of virtual files in /proc:

```
cat /proc/sys/kernel/random/entropy_avail
```

```
hexdump /dev/random
hexdump /dev/urandom
```

> File Globbing

What is it?

How do you prevent it?

Who does it?

> The impossible filesystem! Fun things to do with /proc (why does it exist?)

```
cat /proc/meminfo
cat /proc/cpuinfo
cat /proc/cpuinfo | grep bogomips
```

```
cat /proc/meminfo | grep Swap
```

```
cd /proc/self
cat maps
```