ECE 131 Programming Fundamentals

Dr. Daryl Lee

University of New Mexico

File I/O in a nutshell

- Surprise! All I/O is the same—console or file.
- Unix arranges for you three "standard" files: stdin, stdout, and stderr, all normally associated with the keyboard and console.
- `printf("Hello") == fprintf(stdout, "Hello")`
- `scanf("%d", &x) == fscanf(stdin, "%d", &x)`
- How do you handle "real" files?

fopen, fclose

- All files must be opened before use, and closed after use.

```
FILE *fp;
fp = fopen("filename.txt", "w");
if (fp)
{
   // Okay to process the file here
}
else
{
   // Do something about problem
}
fclose(fp);
```

feof

- Once a file is opened, it is processed until “end of file” occurs:

```
while (!feof(fp))
{
   // Process the file here
}
fclose(fp);
```

fopen modes

- “w” for writing
- “r” for reading
- “a” for appending (adding to end of file)
- “r+” for reading and writing existing file
- “w+” for reading and writing new file
- “a+” for reading and appending
- In latter three, BE CAREFUL!!

Passing file names to programs

- `int main(int argc, char *argv[])`
- `argc` is the number of "tokens" on the command line, and the number of `char *` in the array `argv`.
- `argv[0]` is name program is called with