Makefiles

What?

- text files named Makefile that have **rules**
- used with **make** command

Why?

- convenience - specifies how to build a program
- efficiency - only builds what's necessary based on last mod dates and rules

Rules have **target** *(filename)*, dependency file list, list of commands to build the file

<target>: <files the target depends on>
<tab><command(s) for making target> **#must use tab (not spaces)**

Example

```
#simplified p3 Makefile                  #comment start with # and go to the end of line
align: align.o heapAlloc.o               #Rule 1: how to make align EOF
     gcc align.o heapAlloc.o -o align
align.o: align.c                         #Rule 2: how to make align.o ROF
     gcc -c align.c
heapAlloc.o: heapAlloc.c heapAlloc.h     #Rule 3: how to make heapAlloc.o ROF
     gcc -c heapAlloc.c
clean:                                   #Rule 4: remove OFs to build EOF from scrtch
     rm *.o                                  can have multiple commands per rule
     rm align
```

Examples showing order commands are executed based on dependency files

```
$ls
align.c  Makefile  heapAlloc.c  heapAlloc.h
$make
gcc -c align.c
gcc -c heapAlloc.c
gcc align.o heapAlloc.o -o align
$ls
align  align.c  align.o  Makefile  heapAlloc.c  heapAlloc.h  heapAlloc.o
$rm heapAlloc.o
rm: remove regular file 'heapAlloc.o'? y
$make
gcc -c heapAlloc.c
gcc align.o heapAlloc.o -o align
$make heapAlloc.o
make: 'heapAlloc.o' is up to date.
$make clean
rm *.o
rm align
$ls
align.c Makefile heapAlloc.c heapAlloc.h
```