Make

- Used for compiling individual files.

```
gcc -c aredistinct.c
gcc -c main.c
```

'-c' creates an object file. aredistinct.o
main.o

```
gcc aredistinct.o main.o -o prog
```

links two object files to an executable file prog

- Now, if either object file is changed, the link command needs to be run.

↳ This is where Make comes in

- decides when to recompile above code
- Checks modification of object files

↳
```
aredistinct.o : aredistinct.c
	gcc -c aredistinct.c
```

The object file depends on aredistinct.c

↳ if it is newer than object file, below code is ran

```
main.o : main.c
	gcc -c main.c
```

Can use symbols in Make:

```
GCC = gcc
CFLAGS = -g -Wall -Wshadow

aredistinct.o : aredistinct.c
	$(GCC) $(CFLAGS) aredistinct.c
```

```
main.o : main.c
$(GCC) $(CFLAGS) main.c
```

We need to include linkage in our Make

```
GCC = gcc
CFLAGS = -g -Wall -Wshadow
prog : aredistinct.o main.o
    $(GCC) $(CFLAGS) aredistinct.o main.o -o prog


aredistinct.o : aredistinct.c
 $(GCC) $(CFLAGS) aredistinct.c


main.o : main.c
$(GCC) $(CFLAGS) main.c
```

needs to be at top before these