1. (3 points) Consider the following two files:

```c
//aux.c
int fun1() {
    return 5;
}

//main.c
int main() {
    return fun1();
}
```

What line would need to be added at the beginning of `main.c` in order to successfully compile it (Note that you can not create new files)?

2. Consider the following files:

```c
//aux1.c
int fun1() {
    return 5;
}

//aux2.c
//Assume necessary declarations are made here
int fun2() {
    return fun1() + 6;
}
int fun3() {
    return 7;
}

//main.c
//Assume necessary declarations are made here
int main() {
    int temp = fun3() + fun1();
    return temp + fun2();
}
```

a. (2 points) What are the dependencies for compiling `main.c`?

b. (2 points) What are the dependencies for linking `main.c` (be sure to specify if you are assuming that prior compilation steps take place)?

c. (3 points) Consider the following Makefile

```
all: prog1 prog2
progl: libsimple.a prog1help.o prog1.c
    gcc -o prog1 prog1.c prog1help.o -lsimple
prog1help.o: prog1help.c
    gcc -c prog1help.c
libsimple.a: simple1.o simple2.o
```
ar rv libsimple.a simple*.o

simple1.o: simple1.c
gcc -c simple1.c

simple2.o: simple2.c
gcc -c simple2.c

prog2: libsimple.a prog2.c
gcc -o prog2 prog2.c -lsimple

Assume that make all has been called and executed successfully. Now say that simple1.c is updated (say with touch simple1.c). What files would be built if make all was once again executed?