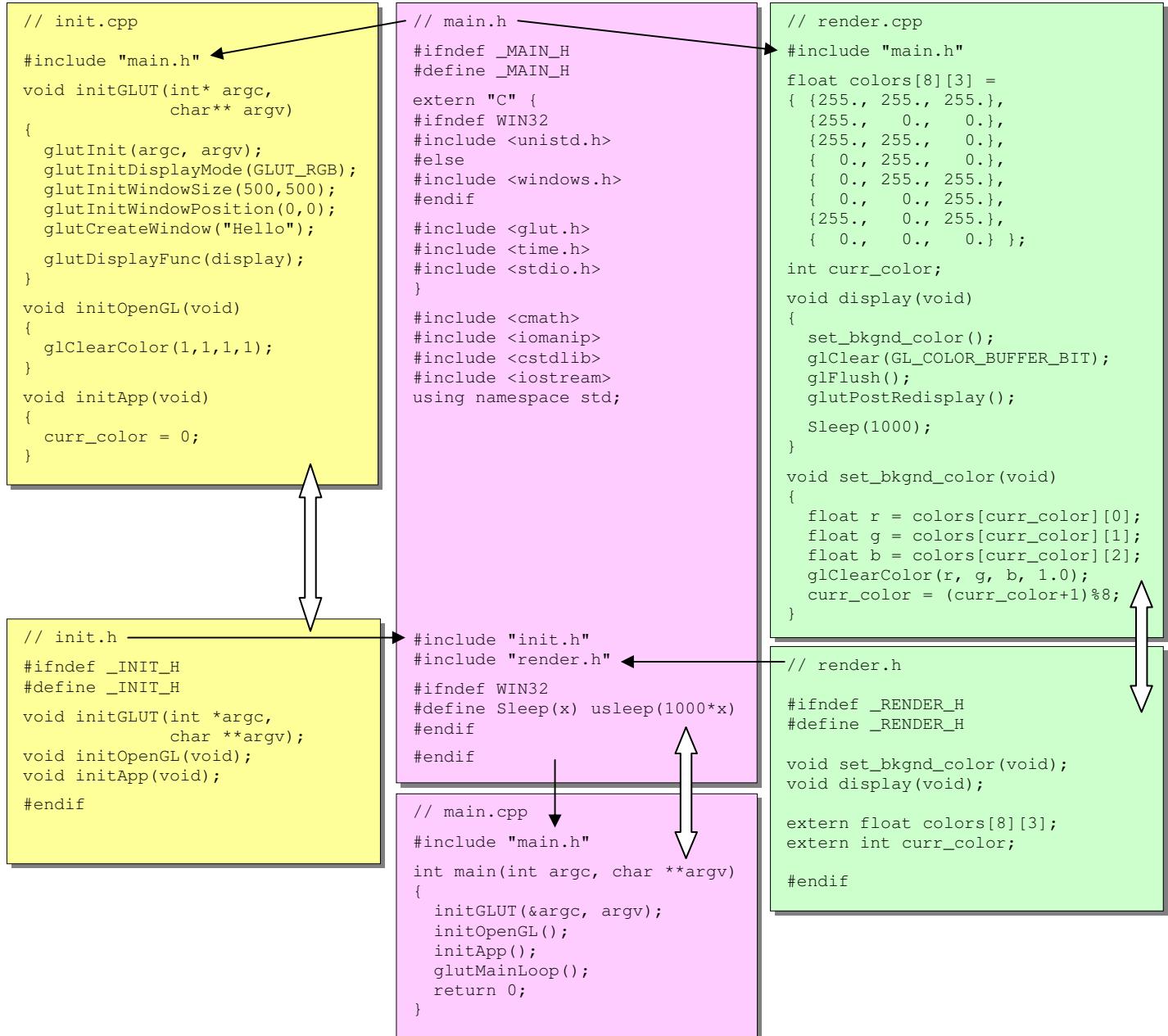


We will program in C/C++, and use **g++** to compile/link. The most recent OpenGL version is 1.5; we will link to an **OpenGL Version 1.3** library.

Our programs will always be multifile, and every **.cpp** file will have a **.h** file with the same name.

We will always have file **main.cpp**.



Makefile you must use. The 1st character in the lines marked with → **MUST** be a **tab**.

```

# this is a comment
INCLUDEDIR      = /usr/local/include
LIBDIR          = /usr/local/lib

OPENALINCLUDEDIR = $(INCLUDEDIR)/AL

XLIBS           = -lX11 -lXmu
GLLIBS          = -lGL -lGLU
GLUTLIB         = -lglut
GLUILIB         = -lglui
OTHERLIBS       = -lpthread -ldl -lopenal

INCLUDEDIRS     = -I$(OPENALINCLUDEDIR) -I.
LIBDIRS         = -L$(LIBDIR) -L.

CC      = /usr/sfw/bin/g++
CCFLAGS = -O3 -c
LIBS    = $(XLIBS) $(GLUILIB) $(GLLIBS) $(GLUTLIB) $(OTHERLIBS)

# all of our OpenGL programs will be named "main"
TARGET = main
##### NO NEED TO CHANGE ANYTHING ABOVE HERE #####
# list of source files
SRCS = $(TARGET).cpp \
→ filename1.cpp \
→ filename2.cpp

##### NO NEED TO CHANGE ANYTHING BELOW HERE #####
# list of header files (one for each .cpp file)
HDRS = $(SRCS:.cpp=.h)

# list of object files (one for each .cpp file)
OBJS = $(SRCS:.cpp=.o)

# rule for making a .o file from a .cpp file
.cpp.o:
→ $(CC) $(INCLUDEDIRS) $(CCFLAGS) $<

# rule for making the final executable
$(TARGET): $(OBJS) $(HDRS)
→ $(CC) $(LIBDIRS) -o $(TARGET) $(OBJS) $(LIBS)

# rule for cleaning up the directory
clean:
→ rm -f $(OBJS) $(TARGET)

# rule for building everything
over:
→ make clean
→ make

```

Ensuring that you:

1. Have a file named `main.cpp`
2. Have a `.h` file with the same name as each `.cpp` file

... then the list of source files is the **only** thing you will have to change in this **Makefile**, from program to program.