

Development Environment Setup

Follow these steps to set up the development environment and run a test C program.

Step #1: Install a Compiler

The compiler you will use is **gcc**. Accessing gcc will be through a command line.

Windows Users

We will be using **Cygwin** to run gcc. Cygwin is a Linux environment with many installable components. We will install gcc core components, audio and graphic libraries.

1. Go to <http://www.cygwin.com> and download **setup-x86_64.exe**.
2. Launch the setup application and click "Next" until you see "Choose a Download Site."
3. Choose a mirror and select "Next."
4. On the "Select Packages" screen, choose "Full" and select the following packages by clicking on the word "Skip" on the corresponding line:
 - a. Search "gcc" and select "gcc-core" and "gcc-g++"
 - b. Search "gdb" and select "gdb"
 - c. Search "portaudio" and select "libportaudio-devel" and "libportaudio2"
 - d. Search "libsndfile" and select "libsndfile-devel" and "libsndfile1"
 - e. Search "glu", and select "libGLU-devel", "libGLU1", "libglut-devel", and "libglut3"
 - f. Search "make" and select "automake", "cmake", and "make".
5. Click "Next" and run the installation. This may take several minutes.
6. Add a Desktop icon for Cygwin, if desired.
7. Run **Cygwin Terminal** and type **gcc --version** at the prompt. If it displays a version number (and not an error) then you're all set.

Mac and Ubuntu (Linux) Users

Mac OS X has a **Terminal** included in the **Utilities** folder. However, you still need to install the developer tools and libraries. To simplify the installation, I use **Homebrew**.

1. Open a web browser, go to <http://brew.sh> and copy the one-line command.
2. In the Finder, Select Go / Utilities and run **Terminal**.
3. Paste the copied Homebrew command into Terminal and press Enter.
4. Let the Homebrew scripts run.
5. Type **brew install portaudio libsndfile**
6. Type **xcode-select --install** and select "Install"
7. Type **gcc --version**. If you see a version number (and not an error) then you're all set.

For Ubuntu/Linux, follow the steps for the Mac OS X installation. Use **Linuxbrew**, which is a fork of Homebrew. Linuxbrew is available at <http://linuxbrew.sh>.

Step #2: Basic Shell Commands

Following are commands you will find useful:

display your current working directory	pwd
list all files in working directory	ls (add -alh for more file details.)
create a new subdirectory in pwd	mkdir <i>directory_name</i>
remove subdirectory if it is empty	rmdir <i>directory_name</i>
change the current working directory	cd <i>directory_name</i>
	cd . (go to parent directory)
	cd (return to home/top directory)

Step #3: Install a Text Editor

Install a text editor. (Don't use the one packaged with your operating system.) Choices include **BBEdit** (<http://barebones.com>), **Atom** (<http://atom.io>), and **Sublime Text** (<http://sublimetext.com>). Save your work in a location that you can easily access using the Terminal. On Mac, this usually means that your files should be somewhere in the home directory of your user account. On Windows, the root directory of Cygwin is C:\Cygwin64\ (or similar).

Step #4: Test your Development Environment

Create a new text file and enter the following text *exactly*:

```
#include <stdio.h>

int main(void) {
    printf("Hello, World!\n");
    return 0;
}
```

Save the file as **hello_world.c**. Now, in your terminal, navigate to the file's directory (using shell commands). Then type:

```
gcc hello_world.c -o hello_world
./hello_world
```

You should then see "Hello, World!" printed to the console.