

WINDOWS (a zip with the following software is attached)

1. Install Cygwin (provided) to "C:\cygwin" (recommended version) or "C:\cygwin64" (to the default location)

- Open the setup-x86.exe (or x86_64)
- Click next
- Choose install from internet and click next
- Choose the default directory and click next
- Click next again
- Choose direct connection and click next
- Choose <http://cygwin.mirror.constant.com> and click next
- In the search bar, type gcc-core
- Click the little plus symbol next to Devel
- Under the New section, click the Skip next to gcc-core (to make it not skip)
- Go back to the search bar and repeat the process for: gcc-g++, gdb, make
- The full name of the components are:
 - gcc-core: Compiler
 - gcc-g++: C++ Compiler
 - gdb: The GNU Debugger
 - make: The GNU version of the 'make' utility
- Click next when all those are selected
- Wait for installation
- Click finish (without shortcuts)

Add 'C:\cygwin\bin' to the Windows Path Environment Variables*

*Control Panel -> System -> Advanced System Settings -> Advanced Tab -> Environment Systems -> Edit the "Path" in System Variables and add the Cygwin directory.

2. Install (extract) devkitARM (provided) to "C:\devkitARM" (by default)

Extract the provided archive (do not use a downloaded installer on Windows)

3. Install (extract) VisualBoyAdvance (provided) to "C:\vba" (by default)

Use the attached version

The executable should be placed in C:\vba (or a directory of your choice)

4. Use your Explorer to determine exactly where the system utilities you just obtained are located. You must know where the following are on your hard drive.

- devkitARM/bin (default: "C:/devkitARM/bin")
- cygwin/bin (default: "C:/cygwin/bin")
- VisualBoyAdvance.exe (default: "C:/vba/VisualBoyAdvance.exe")

5. Edit your Makefile located in your sample project to include the paths in the indicated locations:

DKPATH = Path to devkitARM bin directory

CCPATH = Path to gcc bin directory

VBASIM = Path to vba.exe

6. Edit your Makefile to include your main.c source.

This is where you will list all source files you may include in the future.

SOURCES = main.c

7. Go down to sublime instructions

MAC

1. Make sure you have XCode Tools installed, or the GNU Compiler Collection (GCC) from your Linux distribution's software tool. To check this, open "Terminal" (Applications > Utilities or Applications > Accessories, or you can search for it by pressing Command+Spacebar) and type "gcc" then press enter. You should see the following:

```
lawn-128-61-53-109:~ YourName$ gcc
clang: error: no input files
```

Then try typing "make" and pressing enter. You should see the following:

```
lawn-128-61-53-109:~ YourName$ make
make: *** No targets specified and no makefile found. Stop.
```

If you see "command not found" you need to obtain XCode Command Line tools or reinstall GCC and Make. Sometimes Terminal will automatically ask you to install the Command Line tools, and if it does just follow the instructions to do so. If it does not ask you to install the Command Line tools, type: "xcode-select --install" and press enter.

2. Install devkitARM to "/opt/devkitARM" (by default) (You can also install it wherever you want to, as long as its a directory that won't change)

Make sure you choose the option that says 'OSX'

https://sourceforge.net/projects/devkitpro/files/devkitARM/devkitARM_r45/

3a. Install Homebrew (so that you can install Wine) (So that you can use the windows version of VBA)
Run the following in Terminal:

```
ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

3b. Install Wine

1. Try running "brew install wine"
2. If you get the issue "wine: Xquartz is required to run... etc", run "brew cask install xquartz"
3. Try "brew install wine" again
4. This step will take some time, so just keep checking back every now and then to see if it is done.
5. Brew install winetricks
6. Try to run VBA.exe by typing "wine " (with a space following wine) and then DRAGGING the file "vba.exe" to the terminal window
7. You may get asked to install something called "Gecko", press install
8. VBA will probably crash, at which point you should run the following in Terminal: "winetricks mfc42"
9. Once that is finished, repeat step 6
10. If you still have trouble, ask a TA for help

4. Use your Finder to determine exactly where the system utilities you just obtained are located.

You must know where the following are on your hard drive.

- devkitARM/bin (default: "/opt/devkitARM/bin")
- cygwin/bin (default: "/usr/bin")
- Boycott Advance (default: "/usr/bin/open")

Edit your Makefile to include the paths in the indicated locations:

DKPATH = Path to devkitARM bin directory

CCPATH = Path to gcc bin directory

VBASIM = Path to "/usr/bin/open" by default

5. Edit your Makefile to include your main.c source.

This is where you will list all source files you may include in the future.
SOURCES = main.c

General:

You can now build and run GBA projects using Command Prompt/Terminal. A project folder should contain all c source files and a correctly configured Makefile. The commands "make clean" and "make run" should build a .gba file and run your project.

Running your GBA file with VisualBoyAdvanced will show you a cool animation.

You may now install an IDE for GBA projects. We officially support (and highly encourage) Sublime Text.

Sublime Text Installation:

- Download sublime text from <http://www.sublimetext.com/>
- Copy the text in the Sublime Text Setup Text File
- In sublime, choose Tools -> Build System -> Create New Build System
- Paste the code you just copied to Sublime
- Save the file IN THE DEFAULT DIRECTORY IT GIVES YOU titled GBA
- Go to Tools -> Build System -> GBA (That you just created)
- Inside of the Makefile or main.c, go to Tools -> Build
- You should have a compiled GBA program you can run with a white dot in the center

Submission:

Zip up your **ENTIRE** project folder and submit it. Make sure your submission includes a compiled .gba. Name your submission "Lab01_FirstLast.zip", e.g. "Lab01_CarthOnasi.zip"