CS 2261 Lab 00:
Mac Installation

Provided Files, Folders, and Executables

- .vscode (folder)
    - tasks.json
- Makefile
- main.c
- Dockerfile
- visualboyadvance-m.app

Files to Edit

- tasks.json

Instructions

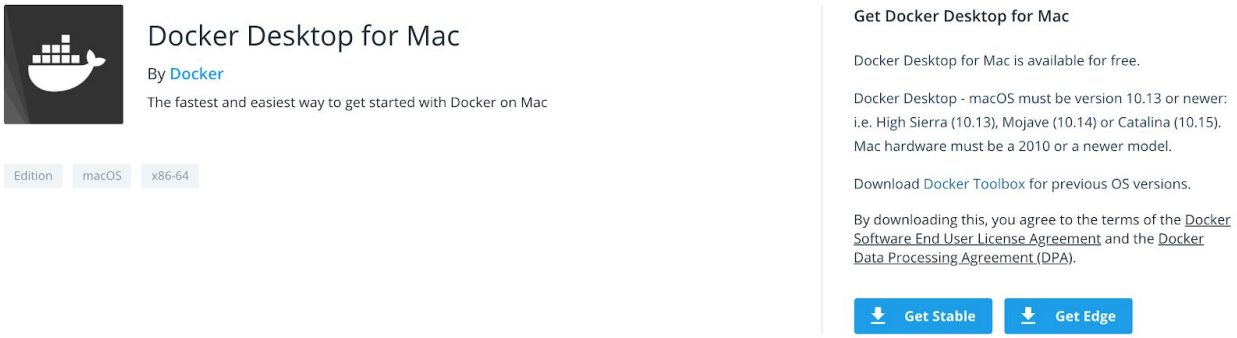
In this lab, you will be installing the software to write, compile, and run GBA games for this class. It is broken up into various parts. If any part does not produce the expected outcome, alert a TA via Piazza and fix the problem before continuing. **It is incredibly important that you read each and every instruction**. Many of your questions will be answered if you read carefully. Do not skim this document! :)

Part 1: Docker Desktop

First, you'll need to download Docker Desktop. In order to do this, you'll need to sign up for Docker Hub. Sign up for Docker Hub here: https://hub.docker.com/signup. You can use your personal or GT email; either is fine.

Once you've signed up, follow the detailed Docker Desktop download instruction here: https://docs.docker.com/docker-for-mac/install/

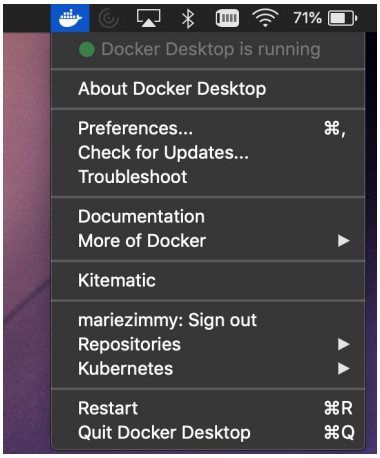
Here, you'll click on the "Download from Docker Hub" button (as seen above) and, granted you're logged into the Docker Hub account you created, you'll be taken to a new page that looks like the following:



Press the "Get Stable" button. In your downloads, you'll see "Docker.dmg".

Now, follow the "Install and run Docker Desktop on Mac" instructions on the Docker Desktop download instructions page (https://docs.docker.com/docker-for-mac/install/#install-and-run-docker-desktop-on-mac). Stop once you reach the “Uninstall Docker Desktop” section. Great, you've got Docker Desktop on your machine! :) Go to your applications, and click Docker.app, as seen below.

You should see, in the top right of your screen, a little icon that looks like the following:



This icon indicates that Docker Desktop is running on your machine! **YOU MUST BE RUNNING DOCKER IN ORDER TO BUILD YOUR CODE**. Otherwise, when you attempt to build your code, you will see the following error:

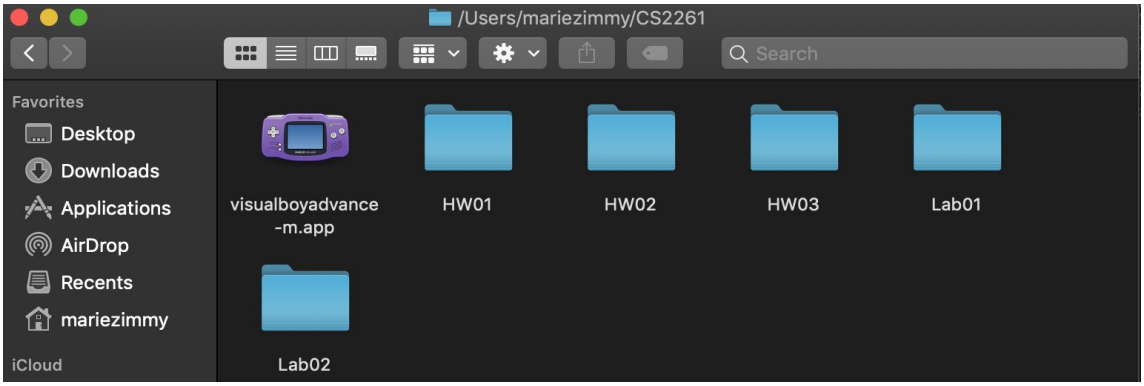
```
Terminal will be reused by tasks, press any key to close it.

> Executing task: docker run --rm -it -v ${PWD}:/gba gba-compiler <

docker: Error response from daemon: dial unix docker.raw.sock: connect: connection refused.
See 'docker run --help'.
The terminal process terminated with exit code: 125

Terminal will be reused by tasks, press any key to close it.
```

If you see this, you'll know you forget to open the Docker Desktop application.

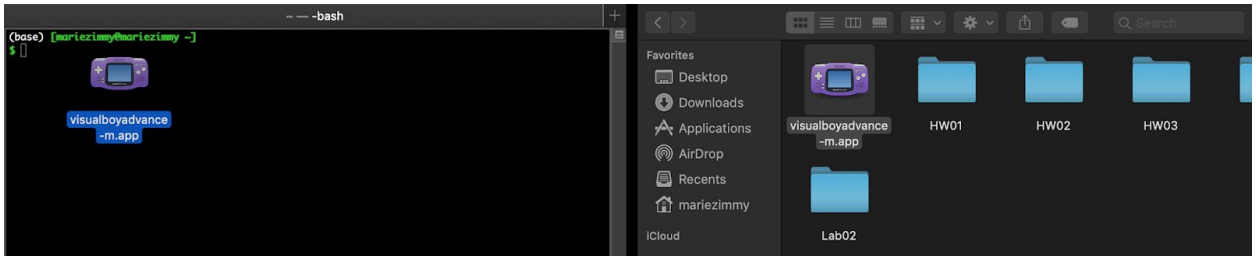
Part 2: VisualBoyAdvance-M

Let's get the GBA emulator setup! VisualBoyAdvance-M is the emulator for this class. If you already have a GBA emulator that you are comfortable with, I still highly recommend you use this one for this class. It has some special features that will come in handy for debugging.

Find the "*visualboyadvance-m.app*" application in the Lab00 folder in a folder parent folder. I recommend creating a parent CS2261 folder that will contain all of your homework, labs, etc. and keeping the "*visualboyadvance-m.app*" in this folder. Lab00 should be in this CS2261 folder, too. Keep a note of the exact path of the "*visualboyadvance-m.app*". In the image below, you'll see a recommended folder setup. The exact path to *"visualboyadvance-m.app*", in my case, is `/Users/mariezimmy/CS2261/visualboyadvance-m.app`



You can verify the directory location of the "*visualboyadvance-m.app*" on your machine by simply opening Finder, opening the CS2261 folder where you placed the "*visualboyadvance-m.app*", and dragging the "visualboyadvance-m.app" icon into the terminal. The output in the terminal is the exact path to the "*visualboyadvance-m.app*". The sequence of images below shows this process.



In my case, as stated before, the exact path is `/Users/mariezimmy/CS2261/visualboyadvance-m.app`. Copy this exact path and paste it into your task.json, on line 9 in between "open" and "./Project.gba". Make sure there is a space after "open" and before "./Project.gba". The (truncated) image below outlines what my tasks.json looks like. Your exact path on line 9 will be different! Remember to save your changes to your tasks.json file.

```
1  {
2      // See https://go.microsoft.com/fwlink/?LinkId=733558
3      // for the documentation about the tasks.json format
4      "version": "2.0.0",
5      "tasks": [
6          {
7              "label": "build run",
8              "type": "shell",
9              "command": "open /Users/mariezimmy/CS2261/visualboyadvance-m.app ./Project.gba",
10             // add the visualboyadvance-m.app exact path to the above line,
11             // in between "open" and "./Project.gba"!
```

Part 3: Visual Studio Code

Visual Studio Code is the text editor of choice for this class. We highly recommend you use this editor. If you do not already have VSCode, you can download it here: https://code.visualstudio.com/download.

Once downloaded, open VSCode, and open the Lab00 folder by selecting File > Open > Lab00.

Open main.c and then hit **cmd+shift+b** to build your project (remember, Docker Desktop needs to be running!). This first compilation process will take some time because a Docker image is being downloaded from Dockerhub (don't worry about the nitty gritty details of this). **All subsequent builds will not take this long** (whew). In the terminal output in VSCode, you should see something like this for the very first time you build:

```
> Executing task: docker run --rm -it -v ${PWD}:/gba aaaronic/gba-compiler:1.0 <

Unable to find image 'aaaronic/gba-compiler:1.0' locally
1.0: Pulling from aaaronic/gba-compiler
9cc2ad81d40d: Already exists
02c01b68baa6: Pull complete
49113593f45f: Pull complete
dc2287d88de6: Pull complete
f8760358be62: Pull complete
Digest: sha256:4fbb0128322187b9bfc4809bfc05c6f6e151dc90523723f2488e8c1ff41b751c
Status: Downloaded newer image for aaaronic/gba-compiler:1.0
/opt/devkitpro/devkitARM/bin/arm-none-eabi-gcc -mthumb-interwork -marm -mlong-calls  -O2 -Wall -pedantic -Wextra -std=c99 -save-temps -D_ROM=Project.gba -D_VBA=C:\U
sers\admin\Desktop\visualboyadvance-m.exe  -c main.c -o main.o
/opt/devkitpro/devkitARM/bin/arm-none-eabi-gcc main.o -specs=gba.specs -mthumb-interwork -marm -mlong-calls  -lm -o Project.elf
/opt/devkitpro/devkitARM/bin/arm-none-eabi-objcopy -O binary Project.elf Project.gba
/opt/devkitpro/tools/bin/gbafix Project.gba
ROM fixed!
```

Once the build is complete, you should see something like the following pop up:

If instead you get an error that that says you cannot open the visualboyadvance-m app, follow the instructions here to allow your Mac to open the application from an unidentified developer: https://support.apple.com/en-gb/guide/mac-help/mh40616/mac. Then, try building again. You

should see the image above.

If you do, congrats, follow the submission instructions :) If not, recall, you need to have Docker Desktop running to build your code! If this is not your issue, notify a TA via Piazza.

Submission Instructions

Zip up the entire sample project folder, including all source files, the Makefile, and everything produced during compilation **(including the .gba file)**. Submit this zip on Canvas.

Name your submission Lab00_FirstnameLastname, for example:“Lab00_MarieZimmerman.zip”.