You need to use your own machine to do the labs for this class. Ideally, you would have a linux machine for doing the lab. But it is also possible to run the labs in other OSs through docker. Check [this](https://docs.docker.com/get-started/) out if you are unfamiliar with docker. Also, If you have trouble finding a machine for doing the labs, please let us know.

There are a couple dependencies you need to build Lab 0 code. You can either install these dependencies natively to your environment, **OR** use the docker image we provide which has all the dependencies installed.

For Lab 0, there are two options for the development environment for this lab. The first is to use a docker container we prepared (see instructions below). Alternatively, if you already have a development environment set up, you can install the dependency natively into your environment (just do something like apt install flex bison).

## Instructions for installing dependencies directly on your machine:

1. Install the dependencies via package manager:

On ubuntu: apt install gcc g++ make flex bison

If you use linux distribution other than ubuntu, swap apt with the package manager on your distribution

1. Once you have all dependencies installed, download the starter code lab0\_starter.tgz and you can proceed to the lab.

## Instructions for using the docker environment:

1. Download and install the docker engine on your machine. See [here](https://docs.docker.com/engine/install/) for instructions
2. Download the two files we shared: lab0\_docker.tar is a docker image, and lab0\_starter.tgz is the starter code for lab0.
3. Untar the starter code: tar -xzf lab0\_starter.tgz in a directory of your choice.
4. Install the docker image into your local registry
   1. Run docker load -i <PATH TO THE lab0\_docker.tar YOU DOWNLOADED>
   2. Check that the image was added correctly. Run docker images and make sure see something with a tag compilers:0.1
5. Running the docker image
   1. Run docker run -it -v <PATH TO WHERE YOU UNTARED lab0\_starter.tgz>:/code compilers:0.1
   2. The option “-v <PATH TO WHERE YOU UNTARED lab0\_starter.tgz>:/code” simply map the directory containing the Lab0 starter code from your host machine into the docker
   3. After the cmd in step a), you should already be inside a docker image. Navigate to /code (the path where the starter code is mapped inside the docker) and start lab0
   4. You can test that your environment is correct by navigating to /code/src and run make