

## Lab1

**Deadline: Due in lab on Sep 14**

### Requirements

First of all, you have to make sure that you have the right version of Ubuntu. In order to do that, you should open a terminal in Ubuntu. Then, type the following command:

```
lsb_release -a
```

It should display Ubuntu 24.04.3 LTS. Make sure that it is this version. Now, make sure that you have the x86-64 architecture by issuing the following command:

```
uname -m
```

It should display x86-64. Note that x86 is different from x86-64, so make sure that you have x86-64 Ubuntu installed. Now, you are ready to start lab1.

### Descriptions

This lab will make you install required packages to perform basic C compilation tool flow. From the prelab, I assume that you have installed an editor of your choice; emacs or vim. Now, we will install gcc, which is the default gcc compiler that we will be using throughout the course. Before we do that, let's update and upgrade your installed packages on your system. You will run the following commands:

```
sudo apt-get update  
sudo apt-get upgrade
```

These will upgrade your packages. *sudo* command in Linux basically performs actions as sudoer, which you can think as an administrator in other operating systems. You do not want to run and develop your code as sudoer as this is a malpractice unless there is a very specific reason to do so. But anything that modifies the system-related software such as your installed compiler, you want to perform it as sudoer. You will only do this if you want to change system related packages in this course.

Now, let's install gcc and other essential tools for this course. We will install other things such as a debugger in the future, but for now, let's install essential tools by:

```
sudo apt-get install build-essential
```

Now, everything should be ready. Now, download the lab1.tar.gz from the course website. I am intentionally using the file from the course website, not github now. This is for you to practice how to transfer files between your host and guest machine. Follow the instructions below:

For Windows: <https://www.isunshare.com/blog/3-ways-to-transfer-files-between-windows-and-virtualbox/>

For Mac: <https://support.cci.drexel.edu/software-courses/utm-virtualization-m1-macs/utm-sharing-folder-linux-guest/>

Now, your file is in your virtual machine, so we can actually untar the directory and compile. We will first have to untar the tar.gz, so do the following command in the directory where you moved lab1.tar.gz file to.

```
tar -vxzf lab1_skeleton.tar.gz
```

If you are getting permission errors, please refer to the following document and change the permission.

<https://help.ubuntu.com/community/FilePermissions>

Now, you should have a directory named lab1\_skeleton. Go into the directory. Let's make sure that there isn't any garbage, so let's clean it up by:

```
make clean
```

We can now compile by:

```
make
```

If you don't get any error, let's run by:

```
make run
```

If you see "Hello World" in your console, you are done for this lab. Check in with the instructor by showing the output in the terminal.

### Restrictions

- You must have x86-64 architecture. Otherwise, you will actually have some features affecting grades in future labs
- I suggest you learn some basic Linux command by visiting: <https://www.hostinger.com/tutorials/linux-commands>

### Grading

For full marks this week, you must:

- (1 point) show your output

### Submission Files

- Nothing to submit for this lab