

# ENGR1510J Lab0

---

Created by 2023 151 TA Team.



This article uses a [\\*CC BY-NC 2.0\\*](https://creativecommons.org/licenses/by-nc/2.0/) license.

## Basic Rules

---

- Pay close attention to Canvas](<https://www.jicanvas.com/courses/917>), mattermost, and [gitea](#). For now, you may not have access to gitea repos, we will set them up later.
- Labs are mandatory, and we also recommend that you attend the lecture. **Please finish the quiz before leaving!**
- Do not use your phone to take pictures of the computer screen.
- Before asking questions, ask the internet for help.
- Ask questions through mattermost and email.
- Do not copy the `$` when working with terminal.

## Mattermost (15 min)

---

Mattermost is an open-source platform for developer collaboration. Secure, flexible, and integrated with the tools you love.

Mattermost(<https://mattermost.com/>)

As an intern, you could discuss anything about our course on mattermost politely. Asking questions will solve your problem quickly as well as help other interns.

## Login

Make sure you have gitea account first. Visit <https://focs.ji.sjtu.edu.cn/mm/> and sign in with FOCS gitea.

## Channel

Channels organize conversations across different topics. They're open to everyone on your team. To send private communications use Direct Messages for a single person or Private Channel for multiple people.

Mattermost has Channels feature to organize conversations across different topics.

There are three types of channels: Public Channels, Private Channels, and Direct Messages.

You can use link to join an public channel, try this link: <https://focs.ji.sjtu.edu.cn/mm/engr151-24fa/channels/town-square>

## Regulations of mattermost

Although we don't monitor your mattermost performance, there are still some regulations to make our course organized.

- Write your name in pinyin (if applicable) as your nickname
- Be polite and patient when asking questions

## Matlab Setup (25-30 min)

---

You can download it from the [official website](#) or from [hex](#).

Keep your computer connected to the Internet, choose login with a MathWorks Account, and click Next. Then select the license with the "Individual" tag.

**Login in with your SJTU email!**

**For macOS users:**

If you are using macOS with an Intel chip, you can directly download and install it from the above link.

If you are using macOS with Apple silicon, you need first install [Java Runtime](#). Currently, [R2024b](#) provides a package for Apple silicon. However, if you install a package for the intel chip, it also works.

## Git/Gitea Setup (50 min)

---

Official Installation Tutorial: [Win](#), [Mac](#), [Linux](#)

### MacOS Git Installation

Turn off SJTU VPN first!

Before you install, you can check whether you have already installed git in your terminal by typing:

```
$ git
```

If the result is command not found, then you have to install git:

1. Install brew (for students in China)

```
$ /bin/bash -c "$(curl -fsSL https://gitee.com/ineo6/homebrew-  
install/raw/master/install.sh)"
```

or you may use: <https://mirrors.tuna.tsinghua.edu.cn/help/homebrew/>

If you are **outside** China, use the following command instead:

```
$ /bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

2. Do an update to test the brew.

```
$ brew upgrade
```

3. Now install git by brew:

```
$ brew install git
```

## Gitea SSH Key Setup

Gitea Link: <https://focs.ji.sjtu.edu.cn/git/>

Profile (Your avator)>Settings>SSH/GPG Keys

First check if you have installed ssh. (Windows users may refer to [https://learn.microsoft.com/zh-cn/windows-server/administration/openssh/openssh\\_install\\_firstuse](https://learn.microsoft.com/zh-cn/windows-server/administration/openssh/openssh_install_firstuse))

Generate key:

```
$ ssh-keygen -t ed25519 -C "xxx@sjtu.edu.cn"
```

Do NOT change anything, type `enter` til it ends.

Print out the file:

```
cat ~/.ssh/id_ed25519.pub
```

*Tips: On MacOS Finder, to show the hidden files/folders, use `Command + Shift + .`*

Since we haven't setup your repos on gitea, you cannot test your key by now. This part will be repeated in lab1.

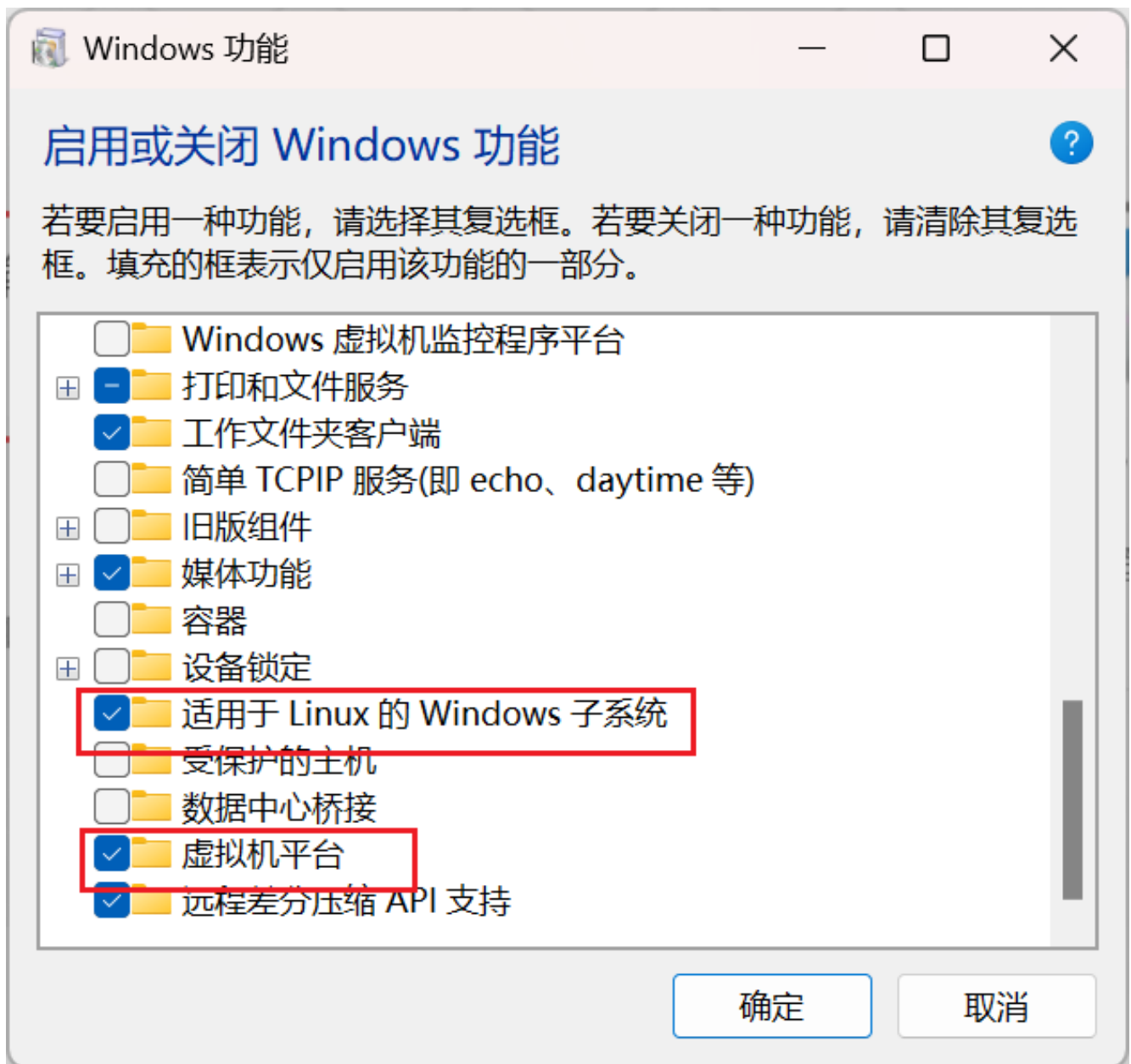
## C/C++ Environment Setup [Not Necessary in this lab] (40 min)

### Windows (using wsl)

#### Requirements

- Windows 10 1709 Fall Creators Update 64bit or later.
- Windows Subsystem for Linux feature is enabled.

1. To enable Linux feature



Then run on `Powershell` with administrator

```
dism.exe /online /enable-feature /featurename:Microsoft-Windows-Subsystem-Linux /all /norestart
```

部署映像服务和管理工具  
版本: 10.0.22621.1

映像版本: 10.0.22621.2134

启用一个或多个功能

[=====100.0%=====]  
操作成功完成。

## Reminder

When you use WSL2, Hyper-V is enabled, which is **NOT** compatible with other vm platforms like VMWare and VirtualBox.

Think twice before using it.

### 2. Check whether your PC support WSL2:

- In cmd run `systeminfo`
- Scroll down to Hyper-V section
- If there are 4 "Yes" OR "A hypervisor has been detected. Features required for Hyper-V will not be displayed." ("已检测到虚拟机监控程序。将不显示 Hyper-V 所需的功能。"), then your PC is OK
- Otherwise it's not OK, please use WSL1 or other vm platform

### 3. Setup wsl2

```
wsl --set-default-version 2
```

`system32` `wsl --set-default-version 2`  
有关与 WSL 2 的主要区别的信息, 请访问 <https://aka.ms/wsl2>  
操作成功完成。

## Case 1: No error

- Just enjoy your WSL2

## Case 2: Error with link <https://aka.ms/wsl2kernel> attached

- Go to <https://aka.ms/wsl2kernel> and download a patch
- start wsl2 again

## Case 3: Error telling you Hyper-V is not enabled

- Go to Control Panel (控制面板) > Programs (程序) > Turn Windows features on or off (启用或关闭 Windows 功能)
- Find Hyper-V

### **Case 3.1: No Hyper-V Settings**

- You should check whether your PC support WSL2. See above.

### **Case 3.2: All 4 settings in Hyper-V can be ticked**

- Tick all settings
- Restart your PC
- Try to start wsl2 again

### **Case 3.3: Some of the settings can't be ticked**

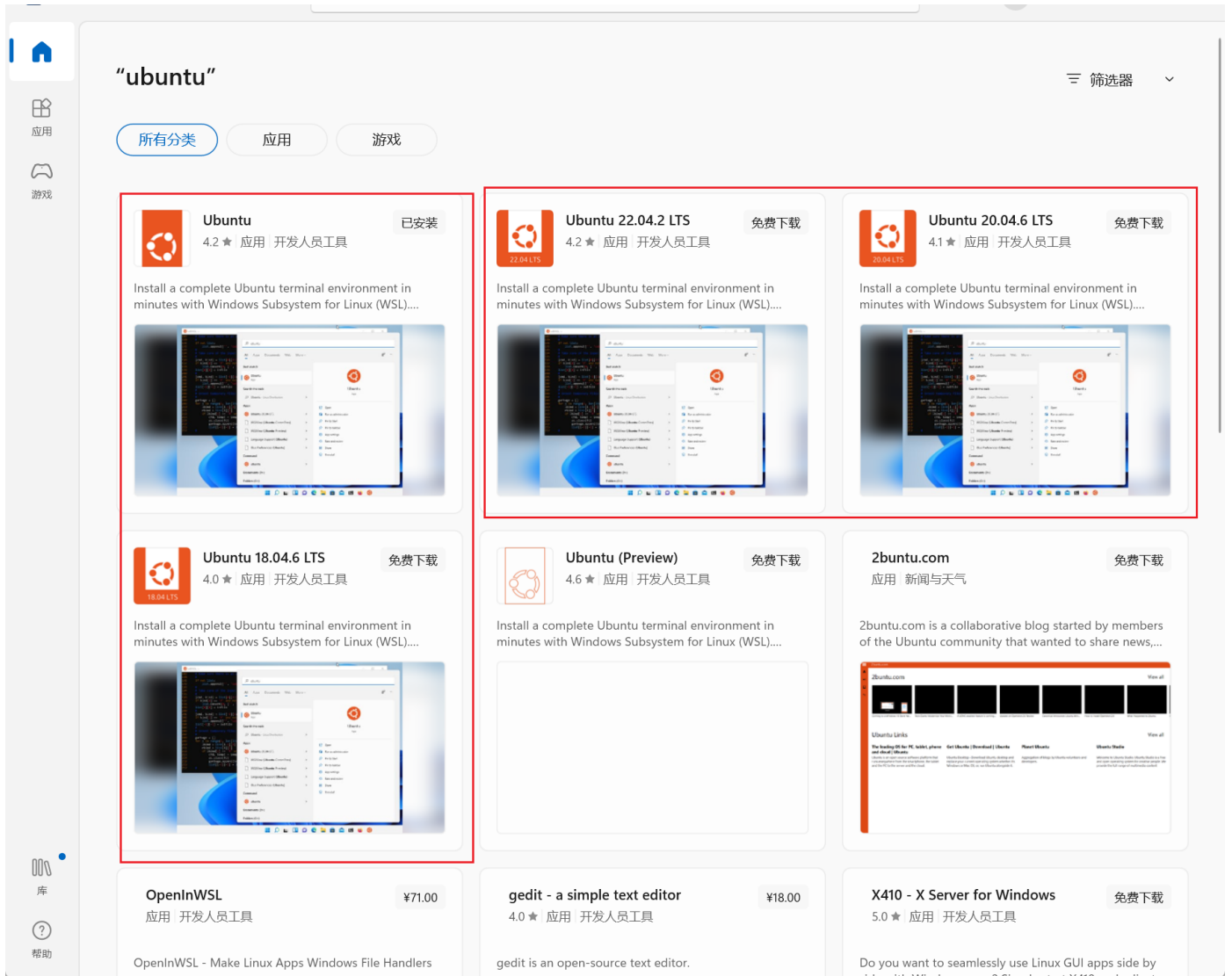
This might be an issue with your hardware. Please refer to section "How to Enable Hardware Virtualization in BIOS" in <https://www.makeuseof.com/windows-11-enable-hyper-v/> and try to enable hardware virtualization. (A former TA's laptop thinkpad has this issue)

### **Case 3.4 Otherwise**

- Search online with error message on your screen (better go to stackoverflow/microsoft doc/github issues)
  - Just use WSL1. It has no big issue.
4. Install Linux (Ubuntu/arch, you only need choose **one**)

## **Ubuntu**

Go to Microsoft Store and search "Ubuntu"



Download one of them. We recommend Ubuntu 22.04.2 LTS.

Open your terminal and choose `ubuntu`.

Setup your user name and password.

```
Installing, this may take a few minutes...
Please create a default UNIX user account. The username does not need to match your Windows username.
For more information visit: https://aka.ms/wslusers
Enter new UNIX username: Jaking
adduser: Please enter a username matching the regular expression configured
via the NAME_REGEX[_SYSTEM] configuration variable. Use the '--force-badname'
option to relax this check or reconfigure NAME_REGEX.
Enter new UNIX username: Jaking --force-badname
adduser: Please enter a username matching the regular expression configured
via the NAME_REGEX[_SYSTEM] configuration variable. Use the '--force-badname'
option to relax this check or reconfigure NAME_REGEX.
Enter new UNIX username: root
adduser: The user 'root' already exists.
Enter new UNIX username: [REDACTED]
Enter new UNIX password: [REDACTED]
Retype new UNIX password: [REDACTED]
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.
```

run

```
$ sudo apt update
$ sudo apt install build-essential
$ gcc --version
```

```
hydraallen@Hydraallen ~$ gcc --version
gcc (Ubuntu 11.4.0-1ubuntu1~22.04) 11.4.0
Copyright (C) 2021 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
```

## Arch

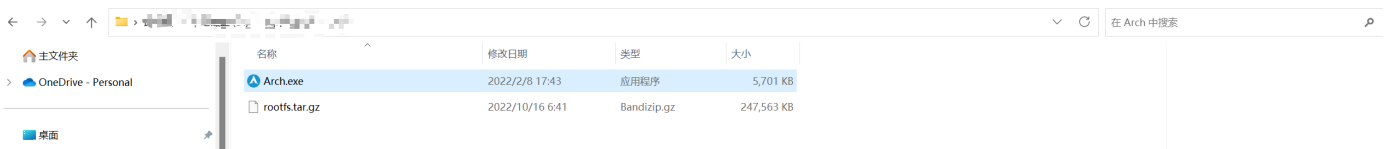
You may refer to the official website.

Here, we demonstrate the first method.

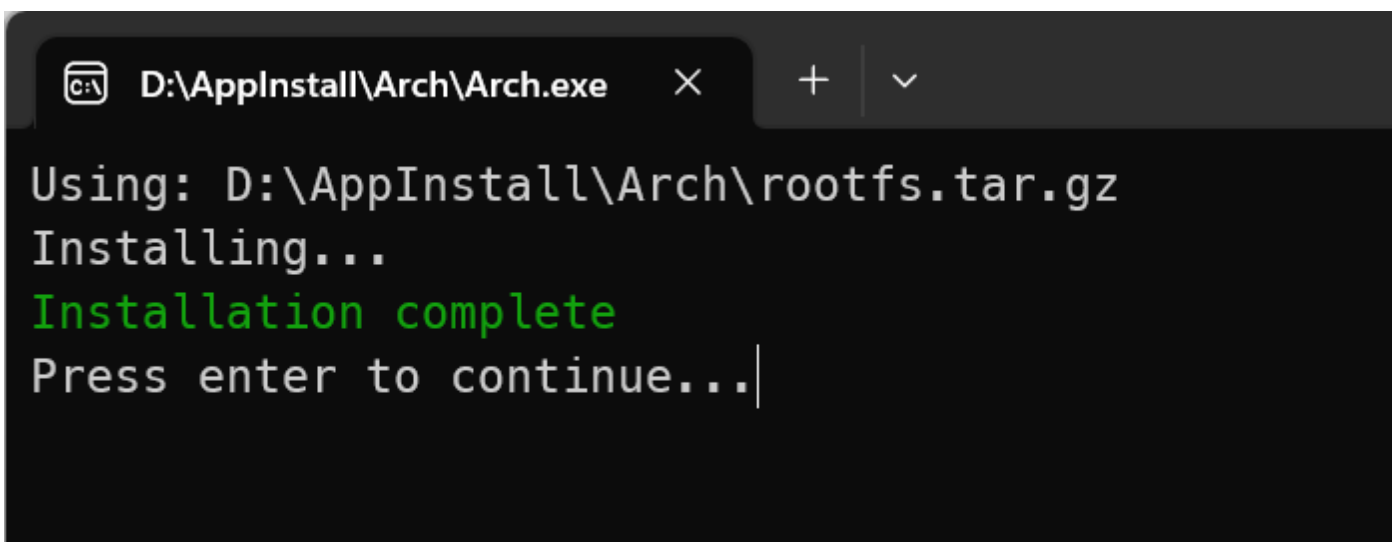
1. [Download](#) the installer zip.

▼ Assets 8		
Arch.zip	242 MB	Oct 16, 2022
ArchWSL-AppX_22.10.16.0_x64.appx	244 MB	Oct 16, 2022
ArchWSL-AppX_22.10.16.0_x64.cer	828 Bytes	Oct 16, 2022
ArchWSL-Online-AppX_22.10.16.0_x64.appx	2.31 MB	Oct 16, 2022
ArchWSL-Online-AppX_22.10.16.0_x64.cer	828 Bytes	Oct 16, 2022
Arch-Online.zip	2.15 MB	Oct 16, 2022
Source code (zip)		Oct 16, 2022
Source code (tar.gz)		Oct 16, 2022

2. Extract all files in zip file to the same directory. Please extract to a folder that you have write permission. For example, `C:\Program Files` cannot be used since the rootfs cannot be modified there.



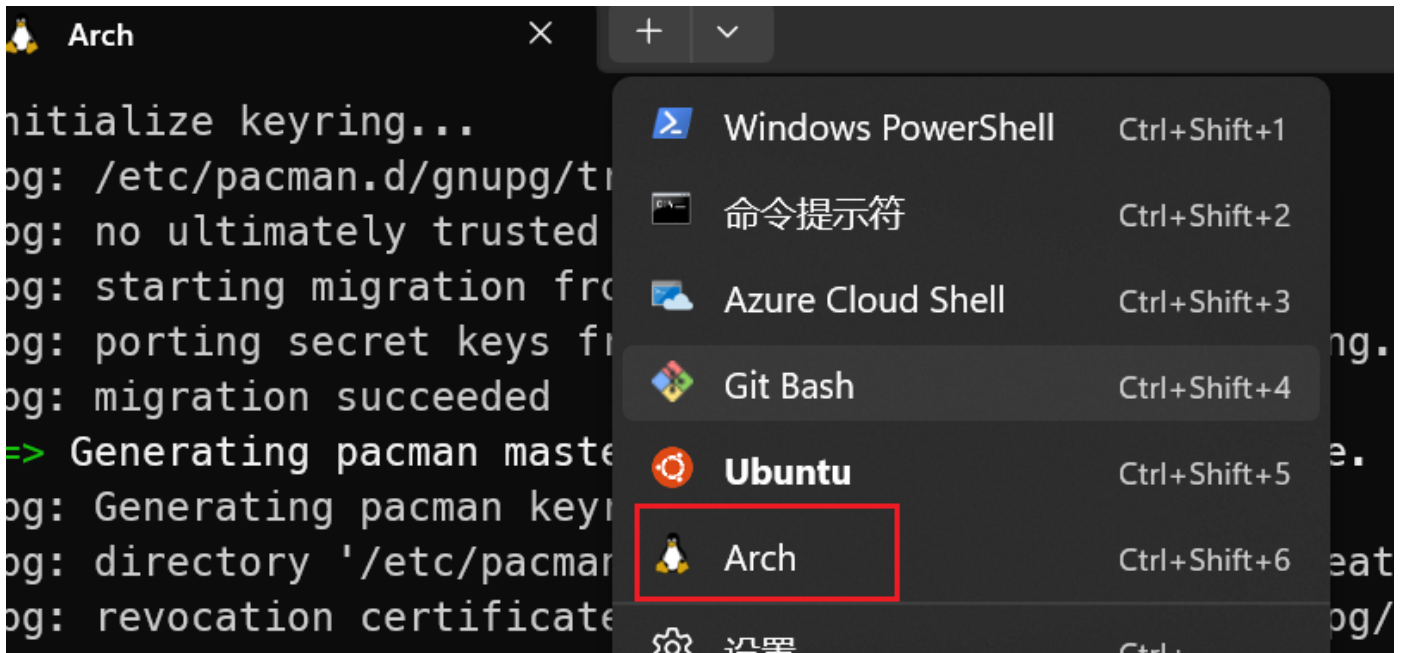
3. Run `Arch.exe` to extract the rootfs and register to WSL





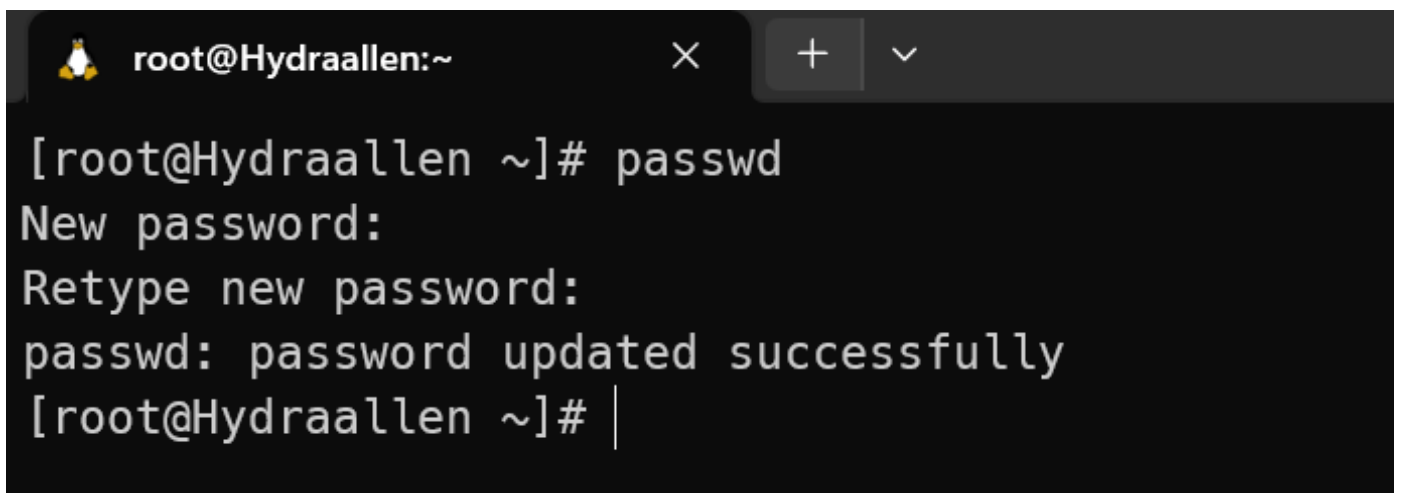
As a side note, the executable name is what is used as the WSL instance name. If you rename it, you can have multiple installs.

4. Open your terminal and choose `arch`.



5. Setting the root password

```
passwd
```

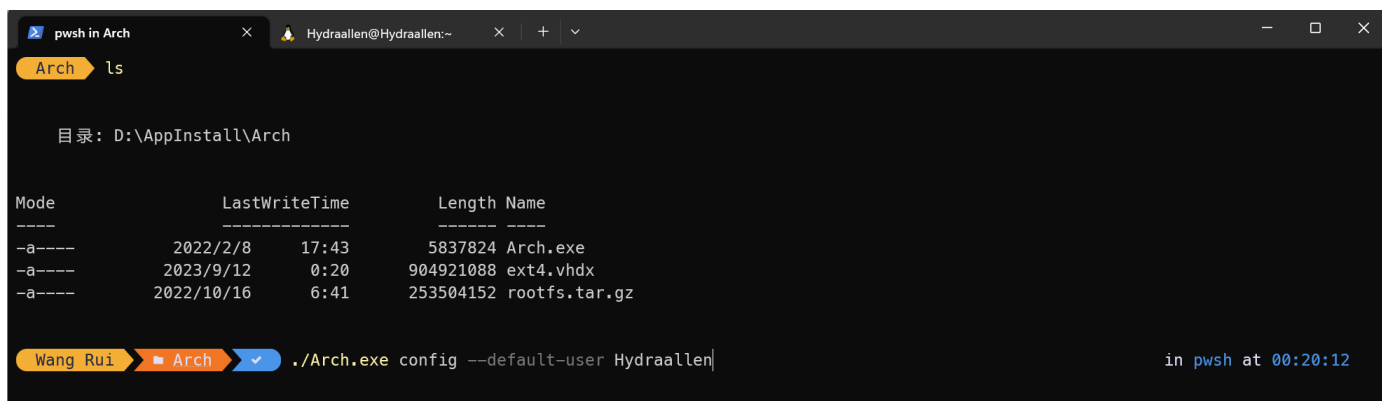


6. Set up the default user

```
$ echo "%wheel ALL=(ALL) ALL" > /etc/sudoers.d/wheel
$ useradd -m -G wheel -s /bin/bash {username}
$ passwd {username}
```

```
[root@Hydraallen ~]# useradd -m -G wheel -s /bin/bash Hydraallen --badname
[root@Hydraallen ~]# passwd Hydraallen
New password:
Retype new password:
passwd: password updated successfully
[root@Hydraallen ~]# |
```

```
$ exit
$ Arch.exe config --default-user {username}
```



```
pwsh in Arch  Hydraallen@Hydraallen:~
Arch ls

目录: D:\AppInstall\Arch

Mode                LastWriteTime         Length Name
----                -
-a----          2022/2/8      17:43         5837824 Arch.exe
-a----          2023/9/12       0:20        904921088 ext4.vhdx
-a----          2022/10/16       6:41        253504152 rootfs.tar.gz

Wang Rui  Arch  ./Arch.exe config --default-user Hydraallen  in pwsh at 00:20:12
```

If the default user has not been changed ([issue #7](#)), please reboot the computer or alternatively, restart the LxssManager in an Admin command prompt. *Please refer to the official documentation.*

## 7. initialize the keyring

```
$ sudo pacman-key --init
$ sudo pacman-key --populate
$ sudo pacman -Sy archlinux-keyring
$ sudo pacman -Su
```

```
Hydraallen@Hydraallen:~$ sudo pacman-key --init

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for Hydraallen:
[Hydraallen@Hydraallen ~]$ sudo pacman-key --populate
==> Appending keys from archlinux.gpg...
==> Updating trust database...
gpg: next trustdb check due at 2023-09-30
[Hydraallen@Hydraallen ~]$ sudo pacman -Sy archlinux-keyring
:: Synchronizing package databases...
   core                               129.3 KiB   41.6 KiB/s   00:03 [#####] 100%
  extra                               8.3 MiB  1882 KiB/s   00:04 [#####] 100%
community                           45.0   B   16.0   B/s   00:03 [#####] 100%
resolving dependencies...
looking for conflicting packages...

Packages (1) archlinux-keyring-20230821-2

Total Download Size:   1.14 MiB
Total Installed Size:  1.62 MiB
Net Upgrade Size:      0.05 MiB

:: Proceed with installation? [Y/n] Y
```

## 8. Install gcc

```
$ sudo pacman -S gcc
```

```
[Hydraallen@Hydraallen ~]$ sudo pacman -S gcc
resolving dependencies...
looking for conflicting packages...

Packages (5) binutils-2.41-3  jansson-2.14-2  libisl-0.26-1  libmpc-1.3.1-1  gcc-13.2.1-3

Total Download Size:   55.42 MiB
Total Installed Size: 231.10 MiB

:: Proceed with installation? [Y/n] |
```

## 9. test gcc

```
$ gcc --version
```

```
[Hydraallen@Hydraallen ~]$ gcc --version
gcc (GCC) 13.2.1 20230801
Copyright (C) 2023 Free Software Foundation, Inc.
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

[Hydraallen@Hydraallen ~]$ |
```

# MacOS

## Install XCode CL Tools

Please follow the following instructions:

First check if you have already installed some compile tools (some macbook preinstalled XCode):

Run this command in your terminal:

```
$ gcc
```

If the result is `clang: error:xxx`, that means you have installed it, you can skip this part.

If the result is something like `zsh: command not found: xxx`, that means you have to install XCode CL Tools.

Run the command to install it:

```
$ xcode-select --install
```

This would take **a longlong time**, so you can install it after lab.

If you meet with some problems in this step, please let us know. (some machines indeed have some problems running this command)

## Install GCC

XCode CL Tools will automatically install clang for you. In this course there are no big differences between clang and gcc. However, some future courses ask you to use gcc rather than clang.

Install gcc by typing:

```
$ brew install gcc
```

Test installation:

```
$ gcc -v
```

If you still cannot directly use `gcc` in terminal, it is because the name of gcc installed by brew is not gcc, it is `gcc-12` (or other versions you installed).

To set gcc and other commands like g++, c++ as default gcc-12 compiler, you need to add some lines in your `~/.zprofile` (if you use zsh).

Copy the following code and paste it into your terminal and run this command. Then everything should be fine.

```
$ echo "alias gcc='gcc-12'\nalias cc='gcc-12'\nalias g++='g++-12'\nalias c++='c++-12'\n\n" >> ~/.zprofile;source ~/.zprofile
```

# Linux

Install everything through your favourite package manager!

*Tips: many Linux distributions include some build tools inside their installation mirror, so you don't have to install them.*

## References

1. [mattermost-user-guide](#)
2. [ENGR1510J Lab0 2022](#)
3. <https://learn.microsoft.com/en-gb/windows/wsl/install> If your windows version is old, please follow <https://learn.microsoft.com/en-gb/windows/wsl/install-manual>