

Be a hacker

楊志璿(SCC)

zxc25077667@pm.me

個人網站

December 4, 2021

Why you join?

窩不知道



whoami

1. [Linux Kernel Module Programming Guide](#) 貢獻者.
2. 繁體中文[第一貢獻者](#) of [Debian Reference](#).
3. 中山大學資工系「作業系統」助教
4. 嘉中電腦社（資研社）107級社長、中山大學資安社創社社長
5. 真hacker [駭客是什麼？](#)、[工程師的價值](#)
6. Senior Backend at [FatBull](#).

良心忠告

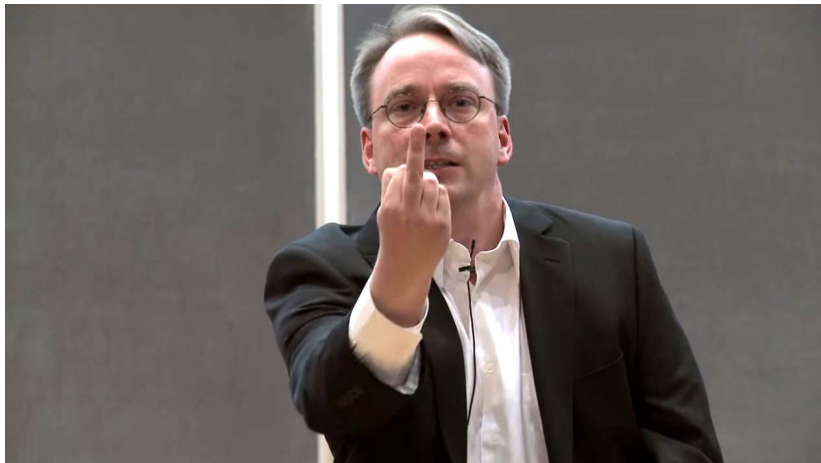
1. 英文很重要(後面都是英文)
2. 自學、多聽多想、敢問、hacker 心態
3. 李琳山院士的建議(2019-2021): [YouTube](#)
4. 建立你們的共筆文化[HackMD](#)
5. **聽不懂請立即**發問於[slido](#)
6. 多參加conference，「認識別人」，最好可以認識「大老」

What you have to learn in this lesson

- ▶ Be a **hacker**, not an user.
- ▶ Learn **Linux** ASAP.
- ▶ Professors are **not** idiots.
- ▶ **Mathematics** is critical but not essential. (which classes)
- ▶ Read the **document**.

Visionary & Engineer

You must see this: [Linus Torvalds](#)



Nvidia

Outline

1. Hacker
2. Linux
3. Mathematics
4. Document
5. Appendix

Hacker

What's hackers



<https://www.youtube.com/watch?v=kst0xNr2deg>

Introduce some BIG people in hackers

- ▶ Richard M. Stallman (RMS)
- ▶ Linus Torvalds
- ▶ Eric S. Raymond
- ▶ Edward Snowden

User



Linux

Partial Linux

<https://dorm.scc-net.tw/>

Do some practices of CLI.

Mathematics

Nope.

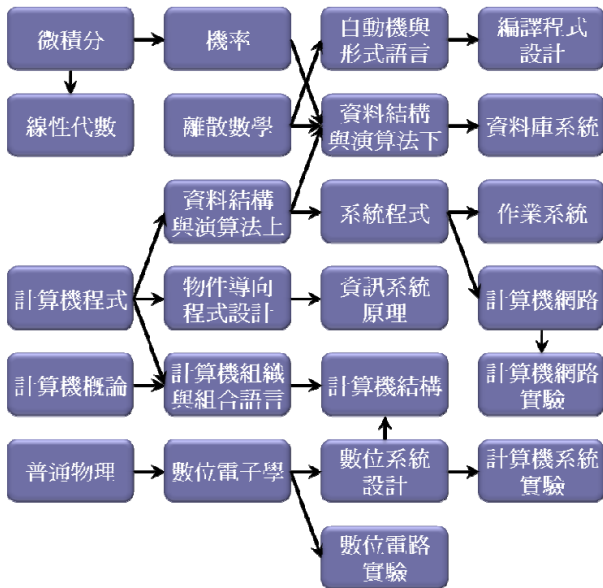
Road map

Teacher: <https://www.csie.ntu.edu.tw/members/teacher.php?mclass1=110>

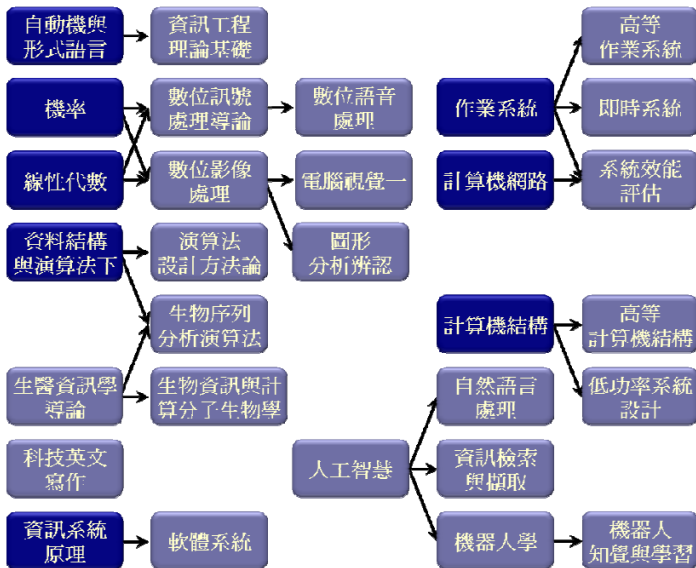
[//www.csie.ntu.edu.tw/members/teacher.php?mclass1=110](https://www.csie.ntu.edu.tw/members/teacher.php?mclass1=110)

Read map: http://coursemap.aca.ntu.edu.tw/course_map_all/class.php?code=9020

大學部必修課程流程



研究所課程流程 (部分)



- ▶ Linear Algebra
- ▶ Calculus
- ▶ Probability and Statistics
- ▶ Algorithm
- ▶ Engineering mathematics

Or be an user . . .

Media

- ▶ Linear Algebra
- ▶ Algorithm
- ▶ Engineering mathematics

- ▶ Probability and Statistics
- ▶ Algorithm
- ▶ Numeracy (Bitwise. . .)

IS, Crypto

- ▶ Algorithm (NP)
- ▶ Linear Algebra
- ▶ Discrete mathematics (Groups)

Network

- ▶ Algorithm (Graph)
- ▶ Probability and Statistics
- ▶ Discrete mathematics (Graph)

Hardware

IDK

You can search it in EE, ECE ...

Gamming Engine

- ▶ Algorithm (Graph)
- ▶ Linear Algebra

Programming Contest

IOI
ICPC

Document

How to read a document?

English

It is why I said that English is important. However, those documents are not hard to read.

(It is a **bad** document if it is hard to read.)

For example:

man ls

```
LS(1)                                User Commands                                LS(1)

NAME
    ls - list directory contents

SYNOPSIS
    ls [OPTION]... [FILE]...

DESCRIPTION
    List information about the FILES (the current directory by default). Sort entries
    alphabetically if none of -cftuvSUX nor --sort is specified.

    Mandatory arguments to long options are mandatory for short options too.

    -a, --all
        do not ignore entries starting with .

    -A, --almost-all
        do not list implied . and ..

    --author
        with -l, print the author of each file

    -b, --escape
        print C-style escapes for nongraphic characters

    --block-size=SIZE
        with -l, scale sizes by SIZE when printing them; e.g., '--block-size=M'; see
        SIZE format below

    -B, --ignore-backups
        do not list implied entries ending with ~

    -c
        with -lt: sort by, and show, ctime (time of last modification of file status
        information); with -l: show ctime and sort by name; otherwise: sort by ctime,
        newest first
```

For example

The linuxserver.io

The [Linux kernel documents](#)

The [fiber package](#)

Write a tiny fiber program

Write a counter program counting the number of 'GET' requests.

Hint: Open your powershell and use the `ssh` to write this program.

```
1 package main
2
3 import "github.com/gofiber/fiber/v2"
4
5 func main() {
6     app := fiber.New()
7
8     app.Get("/", func(c *fiber.Ctx) error {
9         return c.SendString("Hello, World!")
10    })
11
12    app.Listen(":3000")
13 }
```

(Step-by-step)

Appendix

My chances, interviews, and works

1. TTC intern
2. Red-Hat RD (WFH): Give the solution before deadline
3. FatBull senior back-end (WFH) count by working hours:

Why you get the master degree?

- ▶ Can you bypass the junior.
 - ▶ 極短碎碎念
- ▶ I believe that everything will return to the essence.
 - ▶ 駭客是什麼？

My side projects

GitHub

Interviewee

- ▶ Junior
 - ▶ LeetCode(DS/ALGO)
 - ▶ Implement some functional requirements
 - ▶ Basic concepts: socket, Gaussian mask, Any programming language, Design pattern
- ▶ Senior
 - ▶ Working experience: Show your contribution numerically.
 - ▶ Advanced concepts: System design, Hierarchy, Concurrent, Performance, CI/CD

Hack the judge

Why you join?