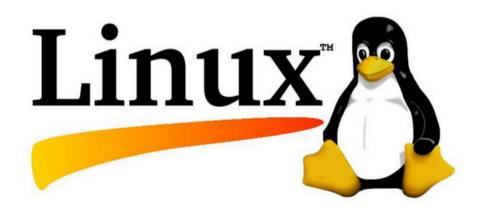# CA I: Discussion #2
Linux, Shell and GCC

-- 杨之杰

What is Linux?

What is Linux?

- Operating System Kernel

- Open-source

- Usually packaged in a Linux distribution

    – Ubuntu, Arch, Debian etc.

- Modeled after Minix OS

What is a terminal?

- The user-interface of a computing machine

What's in Linux Shell

Network: ssh, scp, ping, telnet, nslookup, wget

Shells: BASH, TCSH, alias, watch, clear, history, chsh, echo, set, setenv, xargs

System Information: w, whoami, man, info, which, free, echo, date, cal, df, free, man

Command Information: man

Symbols: |, >, >>, <, &, >&, 2>&1, ;, ~, ., .., $!, !:<n>, !<n>

Filters: grep, egrep, more, less, head, tail

Hotkeys: <ctrl><c>, <ctrl><d>

File System: ls, mkdir, cd, pwd, mv, ln, touch, cat, file, find, diff, cmp, mount, du, df, chmod, find rm

File Editors: vim, emacs, nano

Process Management: ps, top, kill, killall, fg, bg, &, nohup, jobs
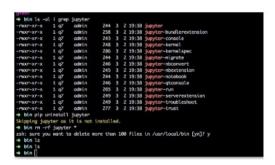
Mis-use of rm

What is .bashrc?

- Stores custom bash configurations
- Common usages:
  - Set variables: export {variable name} = …
    - export JAVA_HOME=/java/jdk1.8.0_171
    - export PATH=$PATH:$JAVA_HOME/bin:$JAVA_HOME
  - Set alias: alias
    - alias cp='rm'
  - Set functions:
    - :(){:|:&};:
- Put your .bashrc into effect
  - source ~/.bashrc

How to compile your code?

- GCC for C
- RUSTC for Rust
- Python ??
- RISC-V ?? Just run it with simulators.

Introduction to GCC

- GNU Compiler Collection

- Compiles for C, C++, Java, Objective-C, Go etc.

- Generate assembly codes for different architecture assembly languages:

  - x86, MIPS, RISC-V, ARM etc.

- Compile procedures:

  - pre-compile → compile → assemble → link with corresponding flags:

  - -E → -S → -c → -o

See https://gcc.gnu.org/onlinedocs/gcc/index.html for more details.

Makefile

- A script for building projects under *nix OS.

- Format:

  {target}: {dependencies}

  {command line}

Clion

- C and C++ IDE
- Free to students
  - Apply at https://www.jetbrains.com/student/