

ITP20004 – Open-Source Software Labs

# Linux Environment

---

***Charmgil Hong***

charmgil@handong.edu

Spring, 2023

Handong Global University



# Announcements

- Weekly schedule

Week	Mon	Week	Thur
1	Course overview, motivation, administrivia	1	CPR: C Programming Reinforcement - Functions
2	Computer organization and Linux environment (1)	2	CPR: C Programming Reinforcement - Strings
3	Computer organization and Linux environment (2)	3	CPR: C Programming Reinforcement - User-defined types, and memory allocation
4	Basic Linux commands + Writing code on Linux (vim)	4	Getting started with Linux / Hands-on Linux command-line tools
5	More Linux commands	5	CPR: C Programming Reinforcement - Understanding compilation and build process
6	Project management (1) <b>Proj 1 출제</b>	6	Project management (2)
7	-	7	Project: BASIC interpreter (2 periods) <b>Project 1</b>
8	Midterm exam	8	Proj 1 due
9	CPR: C Programming Reinforcement - Accessing files and c	9	Debugging with GDB + Unit testing with gtest
10	Code review GNU utilities	10	Writing an application in C
11	Computer network basics	10	Linux network commands <b>AWS 가입 - lightsail</b>
12	Linux machine as a server + Web services	11	Service launching <b>lab problem + AWS 가입해지</b>
13	Project: Text-based Game	13	Github and open-source community <b>Project 2</b>
14	Using Github	14	Socket programming
15	Project: Multi-user game	15	Project: Multi-user game <b>Project 3</b>
16	Final exam	16	

# Announcements

---

- For each lab
  - Before a lab, **every student** submits a pre-lab report (worksheet-type assignment) – **individual work**
  - After a lab, **each team** sees and reports to the TA with the results – **team work**

# Announcements

---

- Team assignment for Weeks 3-5

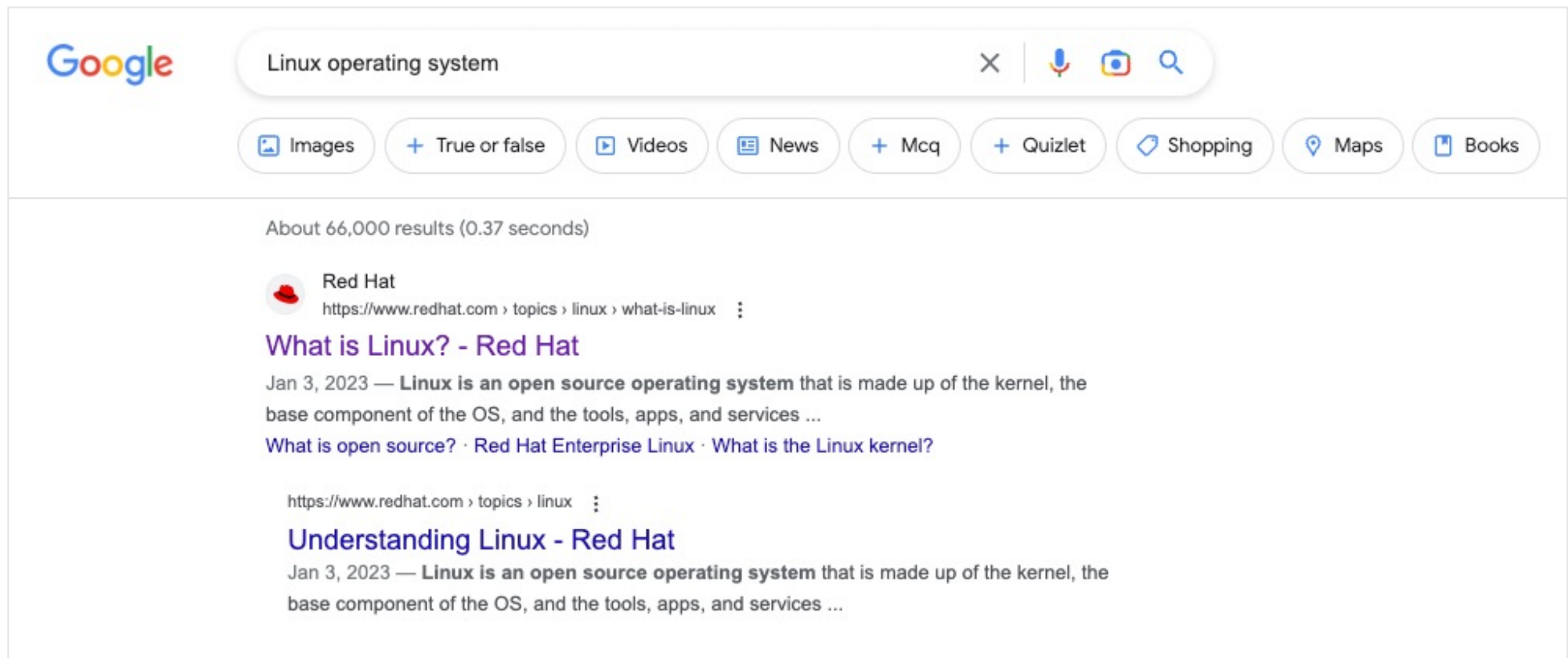
학번	이름
18	마석재
20	이준형
20	김가현
18	김두환
20	정성호
22	곽철호
22	이채연
21	이선환
18	최정겸
22	윤유원
21	조유진
20	정지원
20	나예원
20	비보시놉 아잣
20	김유겸
20	김승환
21	최지안
18	현승준

학번	이름
18	송민준
21	김연희
20	유승준
22	소종현
22	반대준
19	이지명
22	이온유
21	사우지아 유인
21	송영은
21	서준예
18	임건호
22	황찬영
18	박현우
20	윤예람
20	이상현
20	송산
20	방석민
17	김홍찬

# Last Lecture: A Flood of Jargons

---

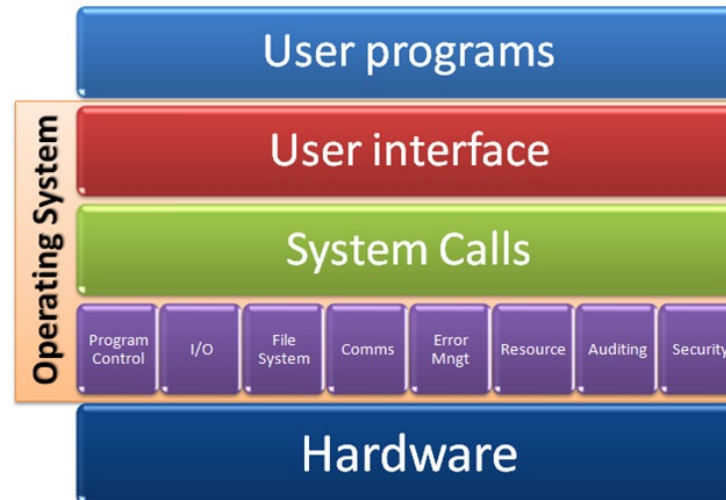
- “Linux is an open-source operating system”



# Last Lecture: Operating System

---

- System software that acts as an **interface** between the user programs and the computer hardware
  - **Manages** computer hardware and software **resources**
  - Provides **common services** for computer programs
  - **Supports communications** among the user programs



\* Image src: <https://www.cs.bgu.ac.il/~spl181/index.php?page=Runtime-environments>

# Last Lecture: Linux

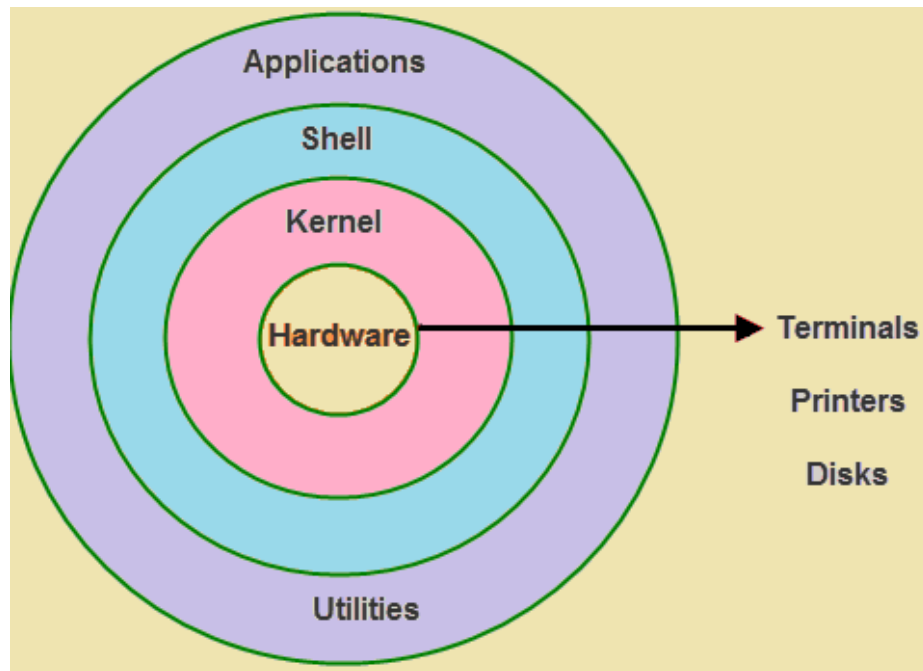
---

- Linux is a *family* of free and open-source software operating systems based on the **Linux kernel**
- Key components
  - Linux kernel: The operating system core
    - Talks to the hardware and software
    - Resource control – Process/memory/file system/device management
  - Shell: User interface
    - Command interpreter
    - Offers easier way to launch applications, navigate thru directories, ...
  - Applications
    - Programming tools – *e.g., Gcc, Eclipse*
    - Editors – *e.g., Vim, Nano, Emacs*
    - Productivity – *e.g., Gimp, OpenOffice*

# Last Lecture: Linux

---

- Linux is a *family* of free and open-source software operating systems based on the **Linux kernel**
- Key components



\* Image src: <https://ssd-pqr.medium.com/introduction-b702cc939cae>



# Last Lecture: Open-Source Software

---

- Software products that include **permission to use its source code, design documents, or contents**
  - Source code is released under an *open-source license*, in which the copyright holder grants users the **right to study, change, and distribute** the software to anyone and for any purpose (Laurent, 2008)
- This is in contrast to *proprietary software*
  - The software is under restrictive copyright
  - The source code is usually hidden from the users

# Last Lecture: In the Beginning...



Richard M. Stallman

Xerox 9700



**UNIX®**



\* Image src: <http://alexb.org/2015/04/21/unix/>; <https://medium.com/@amogh/the-story-of-open-source-so-far-bfcb685d85a4>

# Last Lecture: GNU is Not Unix (GNU)

---

- The GNU project (1983)
  - GNU: GNU is Not Unix (*recursion!*)
  - Build a complete suite of UNIX-compatible non-proprietary software system
- Free Software Foundation (1985)
  - Released a collection of some software
    - Text editor
    - Shell
    - Compiler
  - Did not have an operating system

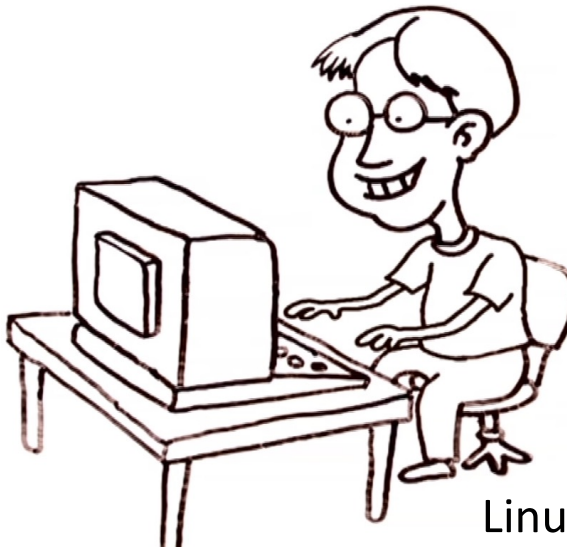


\* Image src: <http://alexb.org/2015/04/21/unix/>; <https://medium.com/@amogh/the-story-of-open-source-so-far-bfcb685d85a4>

# Last Lecture: Birth of Linux



$$+ \text{UNIX}^{\text{®}} = \text{freax (Free + Unix)} \\ = \text{Linux}$$

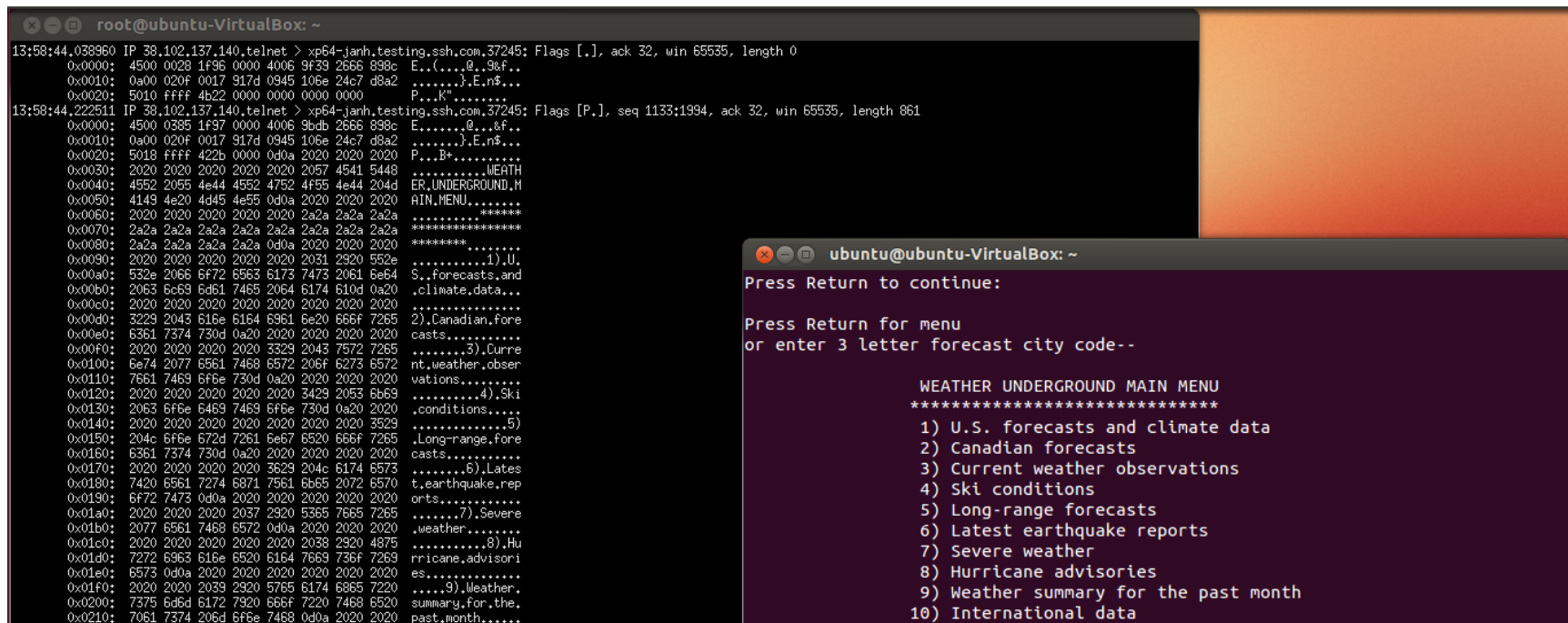


Linus Torvalds  
a student at University of Helsinki

\* Image src: <http://alexsb.org/2015/04/21/unix/>; [https://www.youtube.com/watch?v=5ocq6\\_3-nEw](https://www.youtube.com/watch?v=5ocq6_3-nEw) (by the Linux Foundation)  
<https://www.telegraph.co.uk/technology/2016/08/12/the-first-ibm-pc-was-released-35-years-ago-today---how-it-change/>;

# Last Lecture: Secure Shell (SSH)

- SSH: A network protocol that grants users a **secure way to access** a computer over an **unsecured network**
  - Provides **encrypted data communication** between two computers connecting over an open network (such as the Internet)



The image shows two terminal windows. The left window is a root@ubuntu-VirtualBox terminal showing an SSH connection to xp64-janh.testing.ssh.com. The right window is an ubuntu@ubuntu-VirtualBox terminal showing a telnet session to the same host, displaying a weather service menu.

```
root@ubuntu-VirtualBox: ~
13:58:44.038960 IP 38.102.137.140.telnet > xp64-janh.testing.ssh.com.37245: Flags [.] , ack 32, win 65535, length 0
0x0000: 4500 0028 1f96 0000 4006 9f39 2666 898c E...@...%F..
0x0010: 0a00 020f 0017 917d 0945 106e 24c7 d8a2 .....}.E.n$...
0x0020: 5010 ffff 4b22 0000 0000 0000 0000 P...K".....
13:58:44.222511 IP 38.102.137.140.telnet > xp64-janh.testing.ssh.com.37245: Flags [P.] , seq 1133:1994, ack 32, win 65535, length 861
0x0000: 4500 0385 1f97 0000 4006 9bdc 2666 898c E.....@...&f..
0x0010: 0a00 020f 0017 917d 0945 106e 24c7 d8a2 .....}.E.n$...
0x0020: 5018 ffff 422b 0000 0d0a 2020 2020 2020 P...B+.....
0x0030: 2020 2020 2020 2020 2020 2057 4541 5448 .....WEATH
0x0040: 4552 2055 4e44 4552 4752 4f55 4e44 204d ER_UNDERGROUND.M
0x0050: 4149 4e20 4d45 4e55 0d0a 2020 2020 2020 AIN.MENU.....
0x0060: 2020 2020 2020 2020 2020 2a2a 2a2a 2a2a .....*****
0x0070: 2a2a 2a2a 2a2a 2a2a 2a2a 2a2a 2a2a 2a2a .....*****
0x0080: 2a2a 2a2a 2a2a 2a2a 0d0a 2020 2020 2020 .....*****
0x0090: 2020 2020 2020 2020 2020 2031 2920 552e .....1).U.
0x00a0: 532e 2066 6f72 6563 6173 7473 2061 6e64 S..forecasts.and
0x00b0: 2063 6c69 6d61 7465 2064 6174 610d 0a20 .climate.data...
0x00c0: 2020 2020 2020 2020 2020 2020 2020 2020 .....
0x00d0: 3229 2043 616e 6164 6961 6a20 666f 7265 2).Canadian.fore
0x00e0: 6361 7374 730d 0a20 2020 2020 2020 2020 casts.....
0x00f0: 2020 2020 2020 2020 3329 2043 7572 7265 .....3).Curre
0x0100: 6e74 2077 6561 7468 6572 206f 6273 6572 nt.weather.obse
0x0110: 7661 7469 6f6e 730d 0a20 2020 2020 2020 vations.....
0x0120: 2020 2020 2020 2020 2020 3429 2053 6b69 .....4).Ski
0x0130: 2063 6f6e 6469 7468 6f6e 730d 0a20 2020 .conditions....
0x0140: 2020 2020 2020 2020 2020 2020 2020 3529 .....5).
0x0150: 204c 6f6e 672d 7261 6e67 6520 666f 7265 .Long-range.fore
0x0160: 6361 7374 730d 0a20 2020 2020 2020 2020 casts.....
0x0170: 2020 2020 2020 2020 3629 204c 6174 6573 .....6).Lates
0x0180: 7420 6561 7274 6871 7561 6b65 2072 6570 t.earthquake.rep
0x0190: 6f72 7473 0d0a 2020 2020 2020 2020 2020 orts.....
0x01a0: 2020 2020 2020 2037 2920 5365 7665 7265 .....7).Severe
0x01b0: 2077 6561 7468 6572 0d0a 2020 2020 2020 .weather.....
0x01c0: 2020 2020 2020 2020 2020 2038 2920 4875 .....8).Hu
0x01d0: 7272 6963 616e 6520 6164 7669 736f 7269 rricane.advisori
0x01e0: 6573 0d0a 2020 2020 2020 2020 2020 2020 es.....
0x01f0: 2020 2020 2039 2920 5765 6174 6865 7220 .....9).Weather
0x0200: 7375 6d6d 6172 7920 666f 7220 7468 6520 summary.for.the
0x0210: 7061 7374 206d 6f6e 7468 0d0a 2020 2020 past.month.....
```

```
ubuntu@ubuntu-VirtualBox: ~
Press Return to continue:

Press Return for menu
or enter 3 letter forecast city code--

WEATHER UNDERGROUND MAIN MENU
*****
1) U.S. forecasts and climate data
2) Canadian forecasts
3) Current weather observations
4) Ski conditions
5) Long-range forecasts
6) Latest earthquake reports
7) Severe weather
8) Hurricane advisories
9) Weather summary for the past month
10) International data
```

\* Image source: <https://www.ssh.com/academy/ssh/telnet>

# Lab #1

---

- Task 1 – Handshaking with Linux
- Task 2 – Working with Files and Directories

# Lab #1 Tasks: 1. Handshaking with Linux

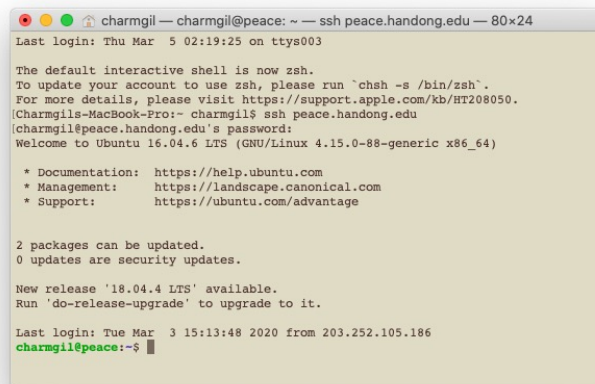
---

- Every individual in the OSSSL course is getting access to the “peace” server, which is installed and maintained by CSEE
  - Use a ssh client software (e.g., Terminal on Mac, Putty on Windows)
    - For detailed information, see last week’s slide deck
  - You need a login credential
    - If it is your first time to use the peace server
      - Your account name is **s + your student id**: E.g., **s21900001**
      - Initial password: **changethispassword**
    - If you have used the peace server before
      - Your **existing password** should work

# Lab #1 Tasks: 1. Handshaking with Linux

## a) Connect to **peace.handong.edu** using SSH

- One can use the built-in Terminal (Mac) or PuTTY (Windows) to log into the peace server



```
charmgil — charmgil@peace: ~ — ssh peace.handong.edu — 80x24
Last login: Thu Mar  5 02:19:25 on ttys003

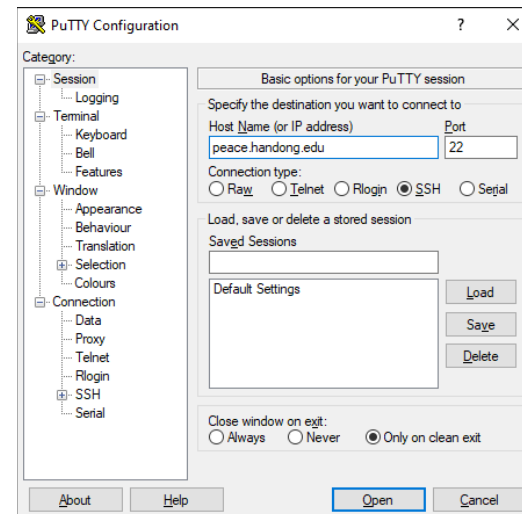
The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(charmgil@MacBook-Pro:~$ ssh peace.handong.edu
(charmgil@peace.handong.edu's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

2 packages can be updated.
0 updates are security updates.

New release '18.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Tue Mar  3 15:13:48 2020 from 203.252.105.186
charmgil@peace:~$
```



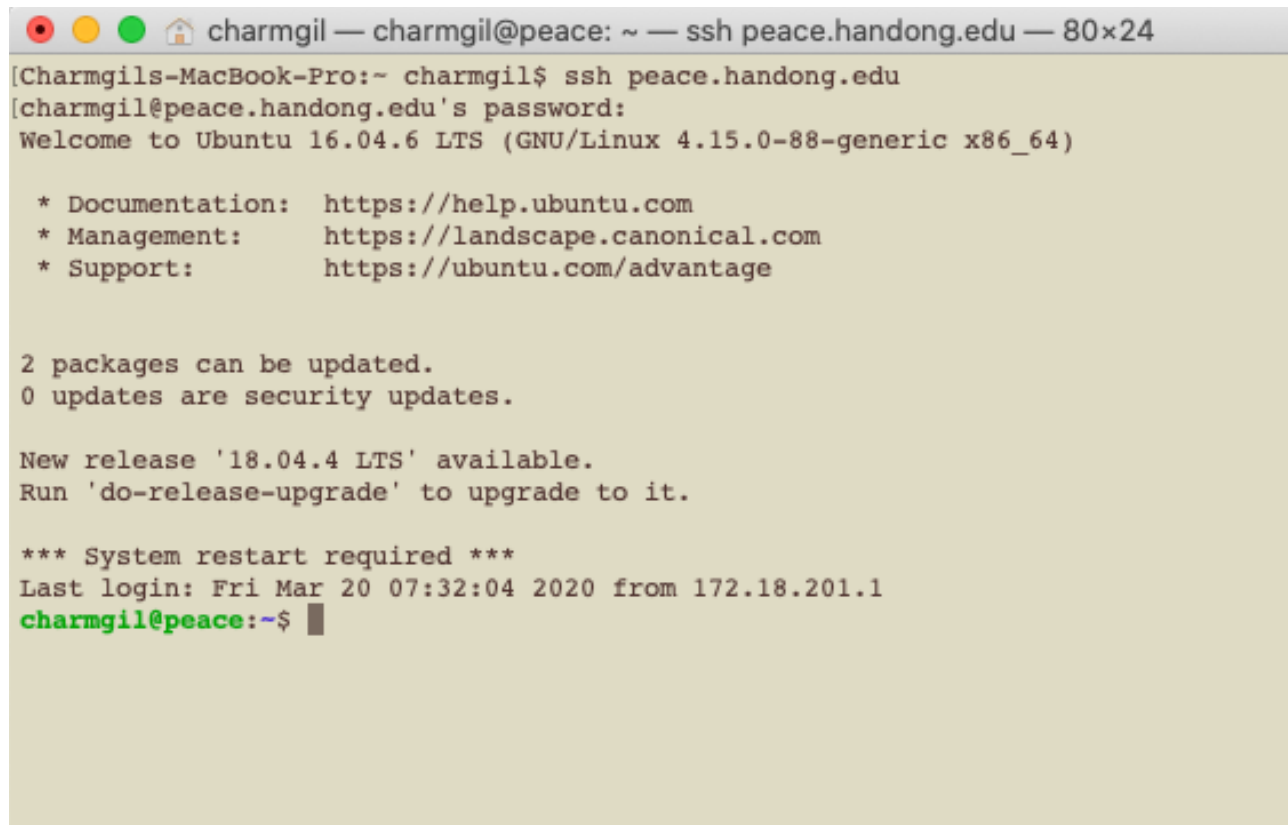


# Lab #1 Tasks: 1. Handshaking with Linux

---

## a) Connect to peace.handong.edu using SSH

- Once you logged in, you will see a screen like below



```
charmgil — charmgil@peace: ~ — ssh peace.handong.edu — 80x24
[Charmgils-MacBook-Pro:~ charmgil$ ssh peace.handong.edu
[charmgil@peace.handong.edu's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

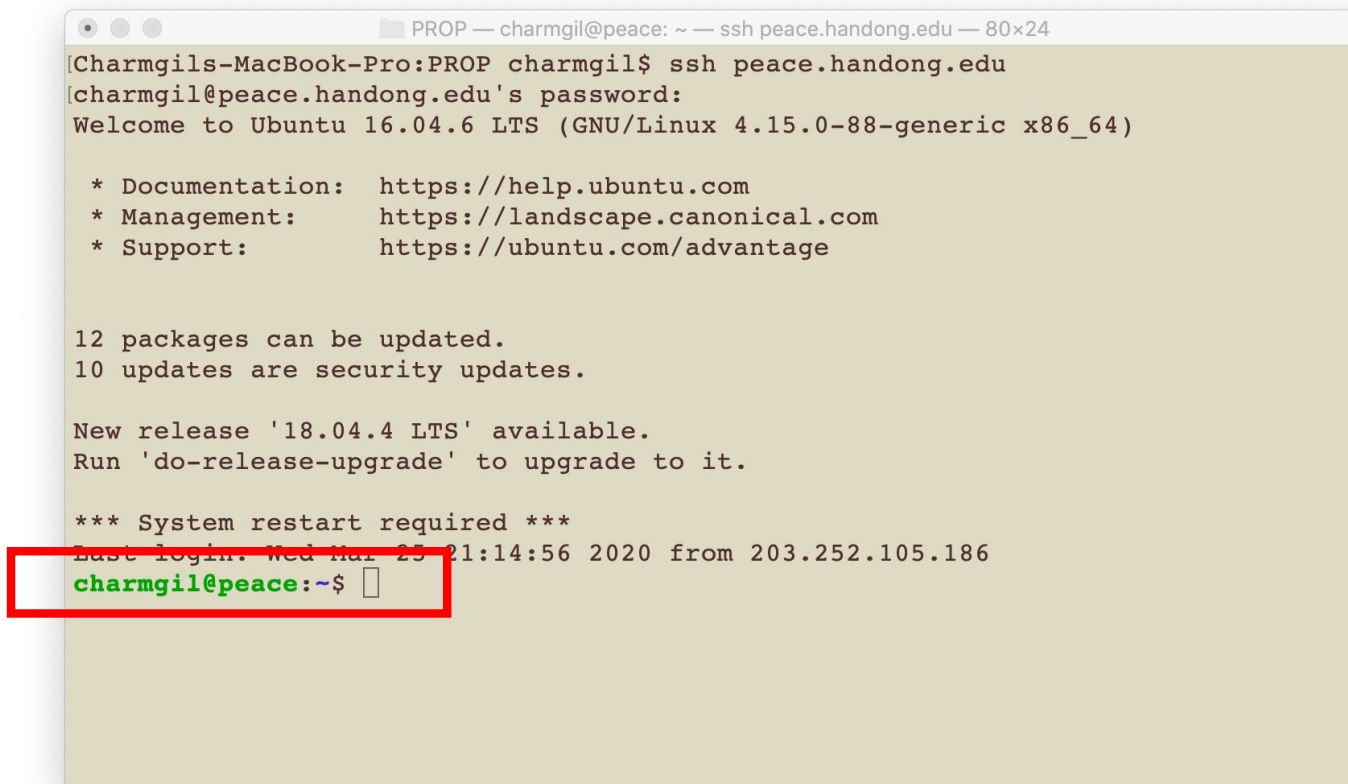
2 packages can be updated.
0 updates are security updates.

New release '18.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Fri Mar 20 07:32:04 2020 from 172.18.201.1
charmgil@peace:~$
```

# Lab #1 Tasks: 1. Handshaking with Linux

- The Bash command line (or Linux command line)
  - Bash: Bourne-Again Shell



```
PROP — charmgil@peace: ~ — ssh peace.handong.edu — 80x24
[Charmgils-MacBook-Pro:PROP charmgil$ ssh peace.handong.edu
[charmgil@peace.handong.edu's password:
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.15.0-88-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

12 packages can be updated.
10 updates are security updates.

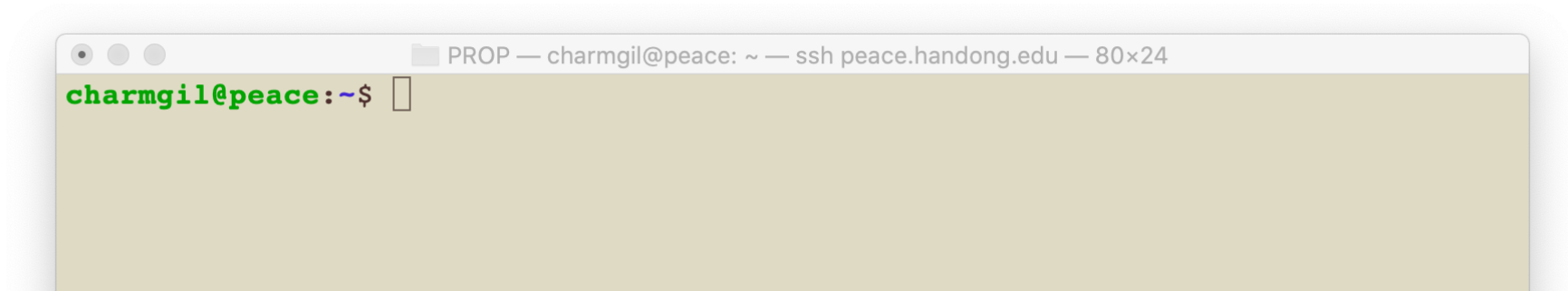
New release '18.04.4 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

*** System restart required ***
Last login: Wed Mar 25 21:14:56 2020 from 203.252.105.186
charmgil@peace:~$
```

# Lab #1 Tasks: 1. Handshaking with Linux

---

a) Connect to `peace.handong.edu` using SSH



- Parse the information that it displays
  - *Account name*
  - *Server name*
  - *Current directory (location in the filesystem)*
  - *Account type (\$: normal user, #: system administrator (root))*

# Lab #1 Tasks: 1. Handshaking with Linux

---

## a) Connect to `peace.handong.edu` using SSH

- The line looks like

```
charmgi1@peace:~$ █
```

is called a **command-line prompt**

- The above text snippet contains several information:
  - `charmgi1`: This block will show your login account name
  - `peace`: The name of the server that you are logged in
  - `~`: Current directory – we will discuss this in next couple of classes
  - `$`: Dollar sign means that you are a normal user
    - If you are a super user (system admin or root), it displays `#`
  - `█`: A blinking cursor – where you can type something in

# Lab #1 Tasks: 1. Handshaking with Linux

---

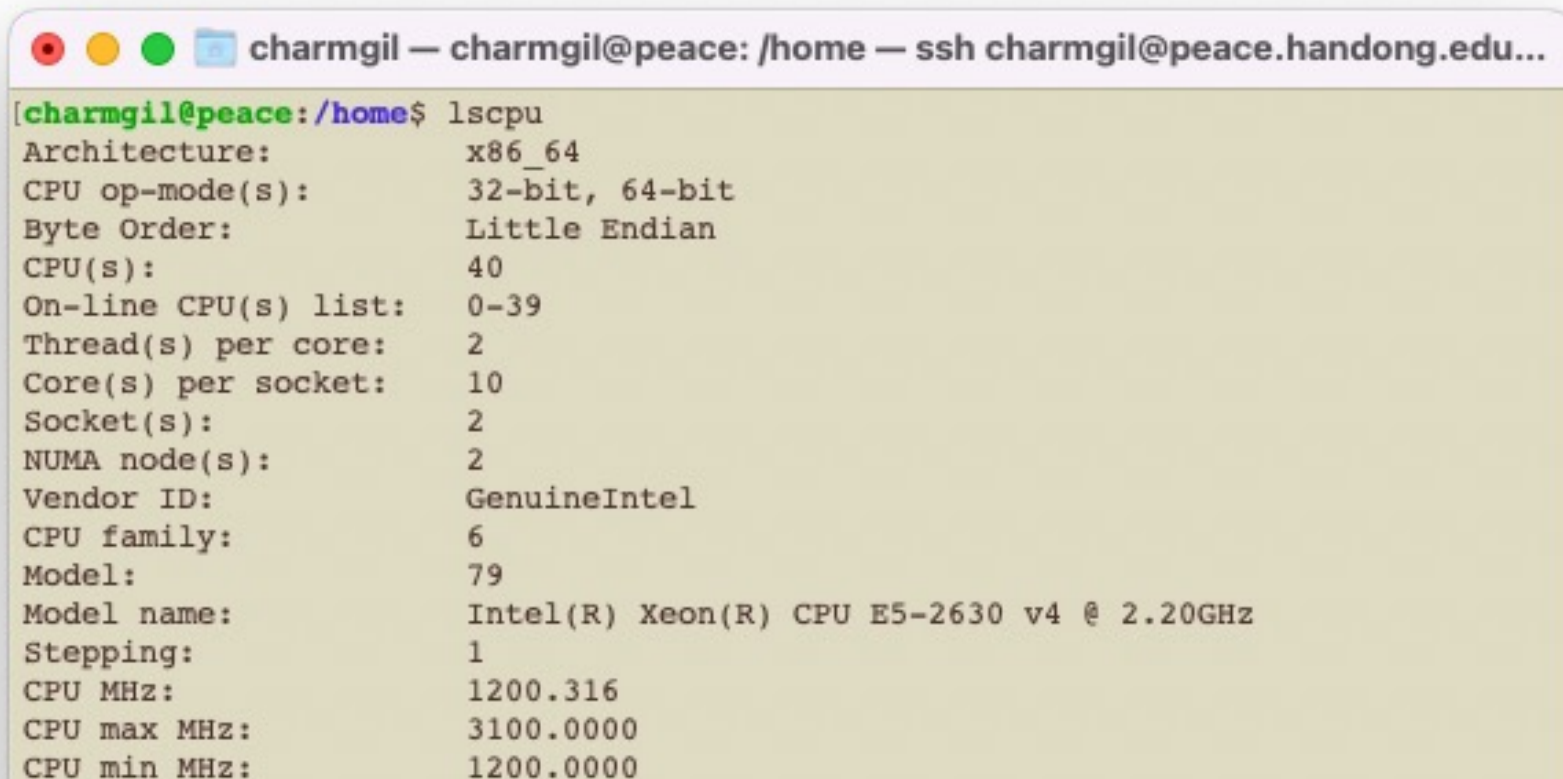
- Wonder when it is?
  - *Try date and cal*



\* Image src: <https://www.discoverlosangeles.com/things-to-do/back-to-the-future-delorean-at-petersen-automotive-museum>  
(Photo courtesy of Petersen Automotive Museum)

# Lab #1 Tasks: 1. Handshaking with Linux


- Want to find the type of process being used?
  - `lscpu` shows the detailed processor information



```
charmgil — charmgil@peace: /home — ssh charmgil@peace.handong.edu...  
[charmgil@peace:/home$ lscpu  
Architecture:          x86_64  
CPU op-mode(s):        32-bit, 64-bit  
Byte Order:            Little Endian  
CPU(s):                40  
On-line CPU(s) list:   0-39  
Thread(s) per core:    2  
Core(s) per socket:    10  
Socket(s):             2  
NUMA node(s):          2  
Vendor ID:             GenuineIntel  
CPU family:            6  
Model:                 79  
Model name:            Intel(R) Xeon(R) CPU E5-2630 v4 @ 2.20GHz  
Stepping:              1  
CPU MHz:               1200.316  
CPU max MHz:           3100.0000  
CPU min MHz:           1200.0000
```

# Lab #1 Tasks: 1. Handshaking with Linux

- Want to find how much memory is available?
  - free shows the system memory usage information (default unit: kilobytes)
    - Mem: (Physical) memory
      - total: Your total (physical) RAM
      - used: memory in use
      - free: memory not in use
      - shared, buffers, cached: show memory usage for specific purposes, these values are included in the value for used.
    - Swap: memory contents that have been temporarily moved to disk (when the physical memory is not enough)



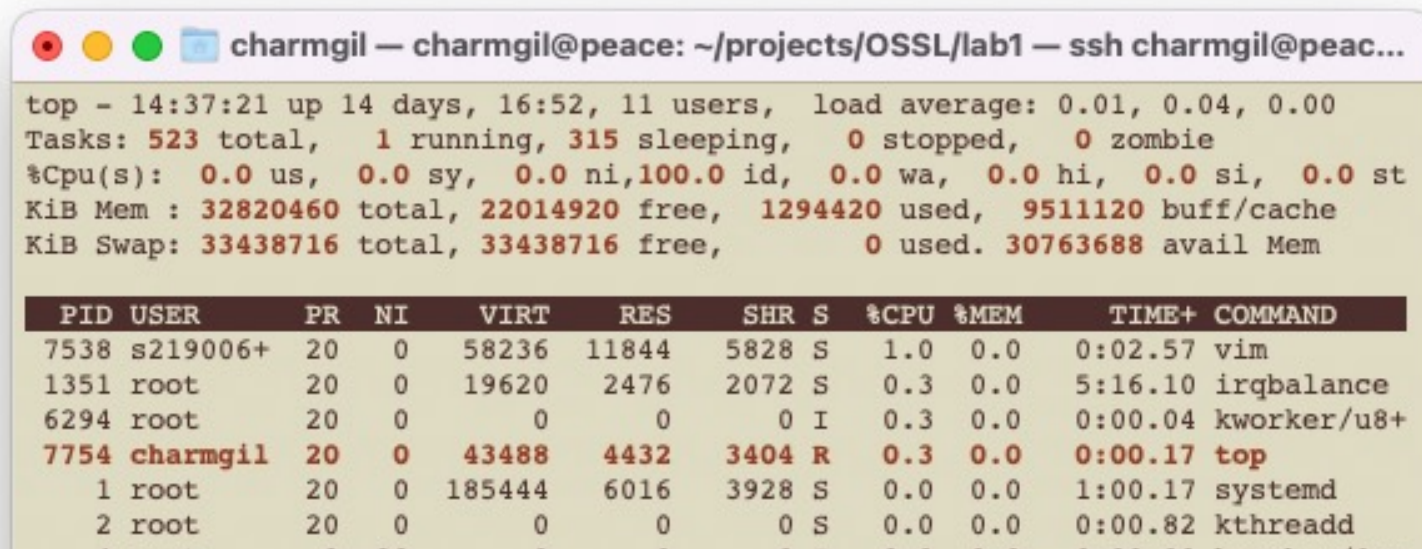
A terminal window titled "PROP — charmgil@peace: ~ — ssh peace.handong.edu — 80x24" displays the output of the `free` command. The output is a table with 7 columns: total, used, free, shared, buff/cache, and available. The rows show memory usage for Mem (Physical) and Swap.

	total	used	free	shared	buff/cache	available
Mem:	32820496	409824	24489912	134184	7920760	31700968
Swap:	33438716	0	33438716			



# Lab #1 Tasks: 1. Handshaking with Linux

- Want to monitor the system usage?
  - top displays the Linux process information
    - It provides a dynamic real-time view of a running system



```
top - 14:37:21 up 14 days, 16:52, 11 users,  load average: 0.01, 0.04, 0.00
Tasks: 523 total,  1 running, 315 sleeping,  0 stopped,  0 zombie
%Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
KiB Mem : 32820460 total, 22014920 free, 1294420 used,  9511120 buff/cache
KiB Swap: 33438716 total, 33438716 free,  0 used. 30763688 avail Mem
```

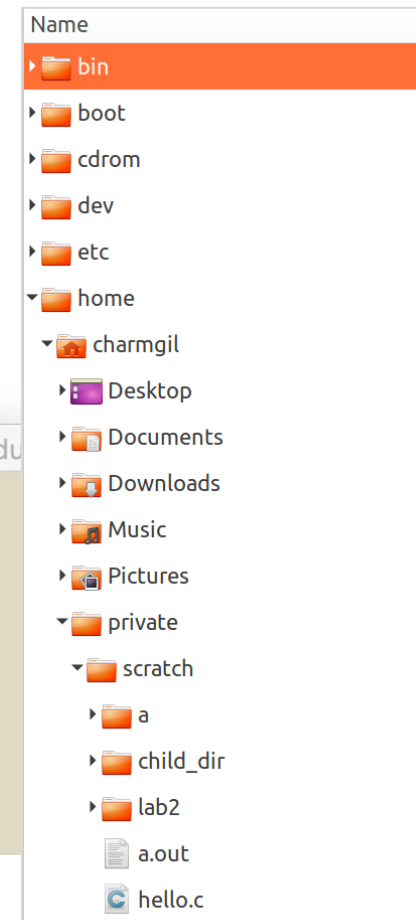
PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
7538	s219006+	20	0	58236	11844	5828	S	1.0	0.0	0:02.57	vim
1351	root	20	0	19620	2476	2072	S	0.3	0.0	5:16.10	irqbalance
6294	root	20	0	0	0	0	I	0.3	0.0	0:00.04	kworker/u8+
7754	charmgil	20	0	43488	4432	3404	R	0.3	0.0	0:00.17	top
1	root	20	0	185444	6016	3928	S	0.0	0.0	1:00.17	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.82	kthreadd



# Lab #1 Tasks: 2. Working with Files and Directories

- Navigate the directory structure
  - Directory: a location for storing files on Linux computers
    - Directory = Folder!

```
PROP — charmgil@peace: ~/projects/code.c — ssh peace.handong.edu
[charmgil@peace:~/projects/data$ cd ~
[charmgil@peace:~$ pwd
/home/charmgil
[charmgil@peace:~$ cd projects/code.c/
[charmgil@peace:~/projects/code.c$ ls
a.out  hello.c
charmgil@peace:~/projects/code.c$
```



# Lab #1 Tasks: 2. Working with Files and Directories

---

- Navigate the directory structure
  - Try the following commands and understand what they do
    - `cd`, `pwd`, `ls`
    - `mkdir`, `rmdir`
  - Read Chapters 4-5 of TLCL (pages 25-53) carefully and exercise all commands in the textbook thoroughly on Peace
    - Write an answer to each question in the hand-out

# Report

---

- Task 1: Show that you can login to the peace server
  - Show the ssh screen that you are on the server
  - Show how to use and explain the screen outputs of `lscpu`, `free`, `top`, `pwd`, `ls`
  - Explain the information you can read from a Linux command line prompt
- Task 2: Answer to the questions that the TA asks to you

# Leftover from the Last Monday

---

- A C program, assembly code, and machine code

# A C Program

- Fibonacci in C

```
#include <stdio.h>

int main(void) {
    int x, y, z;

    while (1) {
        x = 0;
        y = 1;
        do {
            printf("$d\n", x);

            z = x + y;
            x = y;
            y = z;
        } while (x < 255);
    }
}
```

```
% gcc fib.c -o fib
% ./fib
0
1
1
2
3
5
8
13
21
34
55
89
144
233
0
1
1
2
3
5
8
13
21
34
55
89
144
233
```

# C → ASM

- C to assembly

```
#include <stdio.h>

int main(void) {
    int x, y, z;

    while (1) {
        x = 0;
        y = 1;
        do {
            printf("$d\n", x);

            z = x + y;
            x = y;
            y = z;
        } while (x < 255);
    }
}
```

```
% gcc fib.c -o fib
% otool -tv fib
Fib:
(__TEXT,__text) section
_main:
00000000100000f20 pushq    %rbp
00000000100000f21 movq    %rsp, %rbp
00000000100000f24 subq    $0x20, %rsp
00000000100000f28 movl    $0x0, -0x4(%rbp)
00000000100000f2f movl    $0x0, -0x8(%rbp)
00000000100000f36 movl    $0x1, -0xc(%rbp)
00000000100000f3d leaq    0x56(%rip), %rdi
00000000100000f44 movl    -0x8(%rbp), %esi
00000000100000f47 movb    $0x0, %al
00000000100000f49 callq   0x100000f78
00000000100000f4e movl    -0x8(%rbp), %esi
00000000100000f51 addl    -0xc(%rbp), %esi
00000000100000f54 movl    %esi, -0x10(%rbp)
00000000100000f57 movl    -0xc(%rbp), %esi
00000000100000f5a movl    %esi, -0x8(%rbp)
00000000100000f5d movl    -0x10(%rbp), %esi
00000000100000f60 movl    %esi, -0xc(%rbp)
00000000100000f63 movl    %eax, -0x14(%rbp)
00000000100000f66 cmpl    $0xff, -0x8(%rbp)
00000000100000f6d jnl     0x100000f3d
00000000100000f73 jmp     0x100000f2f
```

# ASM → Machine Code

---

- Asm to machine code

```
0x0: ldi    0x1
0x1: sta    [0xe]
0x2: ldi    0x0
0x3: out
0x4: add    [0xe]
0x5: sta    [0xf]
0x6: lda    [0xe]
0x7: sta    [0xd]
0x8: lda    [0xf]
0x9: sta    [0xe]
0xa: lda    [0xd]
0xb: jc     0x0
0xc: jmp     0x3
0xd:
0xe:
0xf:
```

```
0000: 0111 0001
0001: 0100 1110
0010: 0111 0000
0011: 0101 0000
0100: 0010 1110
0101: 0100 1111
0110: 0001 1110
0111: 0100 1101
1000: 0001 1111
1001: 0100 1110
1010: 0001 1101
1011: 1000 0000
1100: 0110 0011
```

# References

---

- Ben Eater. Comparing C to machine language. URL: <https://www.youtube.com/watch?v=yOyaJXpAYZQ>