CSF2600505 Sistem Operasi
CSGE602055 Operating Systems
Week 00: Intro & Review1

Rahmat M. Samik-Ibrahim

University of Indonesia

http://rms46.vlsm.org/2/207.html
Always check for the latest revision!

REV107 24-JAN-2018

Operating Systems 2018-1 (Room 3114 Tue/Thu)
Class: A (10:00-12:00) | B (13:00-15:00) | C (16:00-18:00)

Week	Schedule	Topic	OSC9
Week 00	06 Feb - 12 Feb 2018	Intro & Review1	Ch. 1, 16
Week 01	13 Feb - 19 Feb 2018	Review2 & Scripting	Ch. 2
Week 02	20 Feb - 26 Feb 2018	Protection, Security, Privacy, & C-language	Ch. 14, 15
Week 03	27 Feb - 05 Mar 2018	I/O, BIOS, Loader, & Systemd	Ch. 13
Week 04	06 Mar - 12 Mar 2018	Addressing, Shared Lib, & Pointer	Ch. 8
Week 05	13 Mar - 19 Mar 2018	Virtual Memory	Ch. 9
Reserved	20 Mar - 24 Mar 2018		
Mid-Term	26 Mar - 03 Apr 2018	(UTS)	
Week 06	05 Apr - 11 Apr 2018	Concurency: Processes & Threads	Ch. 3, 4
Week 07	12 Apr - 18 Apr 2018	Synchronization	Ch. 5, 7
Week 08	19 Apr - 25 Apr 2018	Scheduling	Ch. 6
Week 09	26 Apr - 05 May 2018	File System & Persistent Storage	Ch. 10, 11, 12
Week 10	07 May - 16 May 2018	I/O Programming & Network Sockets Programming	
Reserved	17 May - 22 May 2018		
Final	23 May - 26 May 2018	(UAS)	
Deadline	07 Jun 2018 16:00	Extra assignment deadline	

Agenda

About and how to contact the Lecturer

- Rahmat M. Samik-Ibrahim
- Bekerja di Universitas Indonesia: sejak 1984[1].
- Pengguna GNU/Linux: sejak 1994.
- VauLSMorg (vlsm.org): sejak 1996.
- Blog: **rahmatm.samik-ibrahim.vlsm.org/**
  - Blog: 2016/08/panggil-saya-rahmat.html
  - Blog: 2013/10/kumpulan-hal.html
  - Blog: 2011/08/ibu-ke-pasar-membeli-ayam.html
- Facebook: **facebook.com/RMS46F/**
  - Page Only: No need to be "**ADDED**"
  - You can "**Like**" it, "**Follow**" it, or "**Share**" it.
- Kontak/Tanya/Jawab WhatsApp Group **OS181**
  (info +62-881-456-**XXXX**)
  Email (Subject:[HELP]) os181@vlsm.org

[1] MDCCXXXIV — Universitas Goettingen didirikan: **1734**

TOP 10 REALITA

1. Nama saya **Rahmat**. Rahmat nama saya. Kalau bukan Rahmat, bukan nama saya!
2. Jangan datang lebih lambat dari pada Pengajar! Terdapat dua kesempatan untuk menyusul masuk kelas: T+15 menit dan setelah istirahat.
3. Jangan berisik/asyik sendiri dalam kelas dan jangan main "*games*" dan "*chat*" dengan "*gadget*" anda!
4. Jangan lupa mempersiapkan diri untuk berpartisipasi dalam kelas!
5. Jangan lupa membawa selembar kertas (+QRC) untuk membuat memo kuliah!
6. Memo kuliah (+QRC) tersebut yang boleh dibawa saat UTS dan UAS.
7. Jangan curang!
8. Jangan meminjam peralatan selama kuis dan ujian!
9. Jangan menghubungi Pengajar untuk masalah Administratip!
10. Jangan menjadi "*Puss in Boot*"!

Jangan menjadi Puss In Boot

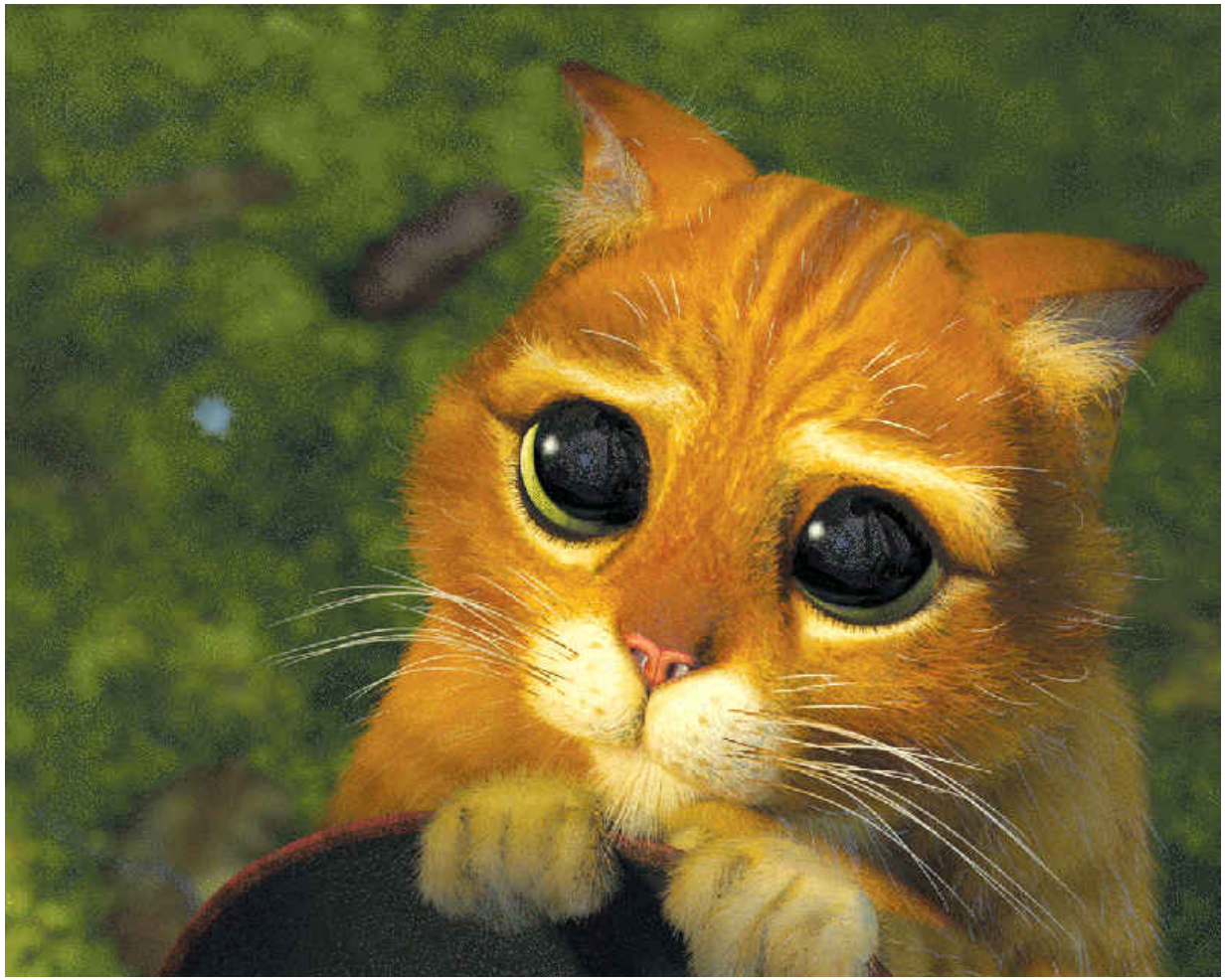


Figure: Ini Puss in Boot[1].

[1] This is a fair use of a DreamWorks/Paramount Picture character.

Goal

Coverage

This is an introduction to a modern operating systems course. It will cover general overview, computer architecture review, operating system overview, IPR, software licenses, GNU/Linux CLI, versioning, scripting, C language overview, protection, security, privacy, gnupg, startup process, I/O, addressing and pointers, memory management, processes and threads, virtual memory, synchronization, mutual exclusion, deadlock, CPU scheduling algorithms, file systems, and I/O programing.

Student-Centered

This course is student-centered where responsibility is in the hands of the students. Students are expected to be prepared for the class meeting.

GNU/Linux

Students will have a thorough understanding of how GNU/Linux provides services by using a Command Line Interface.

Assessment

85 - ... = A	80 - 85 = A-	75 - 80 = B+	70 - 75 = B
65 - 70 = B-	60 - 65 = C+	55 - 60 = C	50 - 55 = D or C[1]
40 - 50 = D	30 - 40 = E	20 - 30 = E	00 - 20 = E

- **4 SKS**: Alokasikan 12 jam per minggu.
- **No Lab — No Task — No Pop Quiz – No Teaching Assistant**.
- **Active Preparation / Participation / Q&A Only**.
- Partisipasi Pra-UTS: 6 set @ 3 points (=18%).
- UTS: 6 set problems @ 6 points (=36%).
- Partisipasi Pasca-UTS: 5 set @ 3 points (=15%).
- UAS: 5 set problems @ 6 points (=30%).
- Extra untuk nilai C keatas: 1 point[1].
- C-2C untuk nilai C-: upto 5 points[1].

[1] Syarat dan Ketentuan Berlaku

Schedule pre MidTerm (UTS)

- Week00 – Intro & Review1 (OSC9-ch01[1] OSC9-ch16 RMS-70).
- Week01 – Review2 & Scripting (OSC9-ch02 Scripting demo-w01[2]).
- Week02 – Protection, Security, Privacy, & C-language (OSC9-ch14 OSC9-ch15 demo-w02).
- Week03 – I/O, BIOS, Loader, & Systemd (OSC9-ch13 demo-w03).
- Week04 – Addressing, Shared Lib, & Pointer (OSC9-ch08 demo-w04).
- Week05 – Virtual Memory (OSC9-ch09 demo-w05).
- MidTerm (UTS) Week 00 — 05.

[1] OSC9: Operating Systems Concepts (9^{th} edition).
[2] Demo Files.

Schedule post MidTerm (UTS)

- Week06 – Concurency: Processes & Threads (OSC9-ch03 OSC9-ch04 demo-w06).
- Week07 – Synchronization (OSC9-ch05 OSC9-ch07 demo-w07).
- Week08 – Scheduling (OSC9-ch06 demo-w08).
- Week09 – File System & Persistent Storage (OSC9-10 OSC9-ch11-OSC9-ch12 demo-w09).
- Week10 – I/O Programming & Network Sockets Programming (demo-w10).
- Final (UAS) Week 06 — 10.

Resources

- Buku Sistem Operasi yang terbit dalam 10 tahun terakhir, umpama: (OSC9) Abraham Silberschatz, Peter B. Galvin, Greg Gagne: Operating System Concepts, 9^{th} Edition, 2013.
- SUP — (ARSIP)(041_Suplemen) Supplement.
- ETC — (ARSIP)(075_ETC-Video) ETC
- (GITHUB) **https://github.com/UI-FASILKOM-OS/os181**
  - (DEMO) — `demos/`
  - (SLIDE) — `pdf/` — **http://rms46.vlsm.org/2/207.html**
- (UJIAN) — **http://rms46.vlsm.org/2/195.pdf - 205.pdf**
- ARCHIVE (Arsip bahan pengajaran): **https://scele.cs.ui.ac.id/course/view.php?id=126**
  - Enrollment key: "11100001"[1].
- BADAK — (BADAK) `/extra/`

[1] Kunci akan berubah secara berkala.

Self Service

- What is your class? A? B? C? D? E? I? M? X?
- Creat project "os181" on your new (or existing) github.com account.
- Check you existing SSO Account (for using badak.cs.ui.ac.id).
- (Week 00) QRCode[1]: "OS181
  CLASS ID GITHUB-ACCOUNT SSO-ACCOUNT SIAK-Full-Name"
- (Weekly) Memo.
- Informasi Kuliah, Arsip Ujian, dan Demo
  - `badak.cs.ui.ac.id:/extra/`
  - `https://github.com/UI-FASILKOM-OS/os181`
  - `https://rms46.vlsm.org/2/195.pdf` — [195.pdf - 205.pdf].
- BASH Account:
  - Virtual Ubuntu: badak.cs.ui.ac.id (SSO)
  - Ubuntu (BYOD)
  - WSL: Windows 10 Subsystem for Linux
  - Cygwin (Windows)

[1] "QR Code" is a registered trademark and wordmark of Denso Wave Inc.

Week 00: Encoding and Decoding a QR Code

```
# OS181:           OS 2018 1st term
# CLASS:           A, B, C, D (reguler), E (Extention),
#                  I (International), M (Matriculation), X (ETC).
# ID:              Student ID (NPM). Eg. "1253757175"
# GITHUB-ACCOUNT: Student's GITHUB acount. Eg. "cicak.git"
# SSO-ACCOUNT:    Student's SSO acount. Eg. "cicak.sso"
# SIAK-FULL-NAME: Student's SIAK name. Eg. "Cicak Bin Kadal"

$ qrencode "OS181 B 1253757175 cicak.git cicak.sso Cicak Bin Kadal" \
     -s 7 -o OS181-B-1253757175.png

$ zbarimg OS181-B-1253757175.png
QR-Code:OS181 B 1253757175 cicak.git cicak.sso Cicak Bin Kadal
scanned 1 barcode symbols from 1 images in 0.11 seconds
```

Quick Response Code (QRC) Code
Check yours with a QRC reader app.

Memo Mingguan

- **WAJIB**: mempersiapkan/mempelajari bahan kuliah minggu terkait.
  - telah memahami garis besar bahan minggu terkait.
  - telah mempelajari jenis soal UTS/UAS yang pernah ditanyakan pada masa lalu.
- Telah mempersiapkan diri dengan membuat memo yang ada *QRC*.
  - Harap **TEST** apakah QRC terbaca dengan aplikasi QRC reader.
  - QRC memo akan di-*scan* di kelas pada saat istirahat kuliah pertama minggu terkait[1].
  - tujuan pembuatan memo ialah sebagai "bukti" telah belajar.
  - ISI memo tidak dinilai!
  - Memo yang baik ialah MEMO yang bermanfaat untuk pembuatnya.
- Ujian:
  - Saat UTS dipersilakan membawa hingga 6 lembar memo ber QRC.
  - Saat UAS dipersilakan membawa hingga 5 lembar memo ber QRC.
  - Memo boleh yang pernah ditulis atau boleh juga membuat memo (ber QRC) yang baru.

[1] kecuali kuliah minggu 00.

Week 00 Memo Example

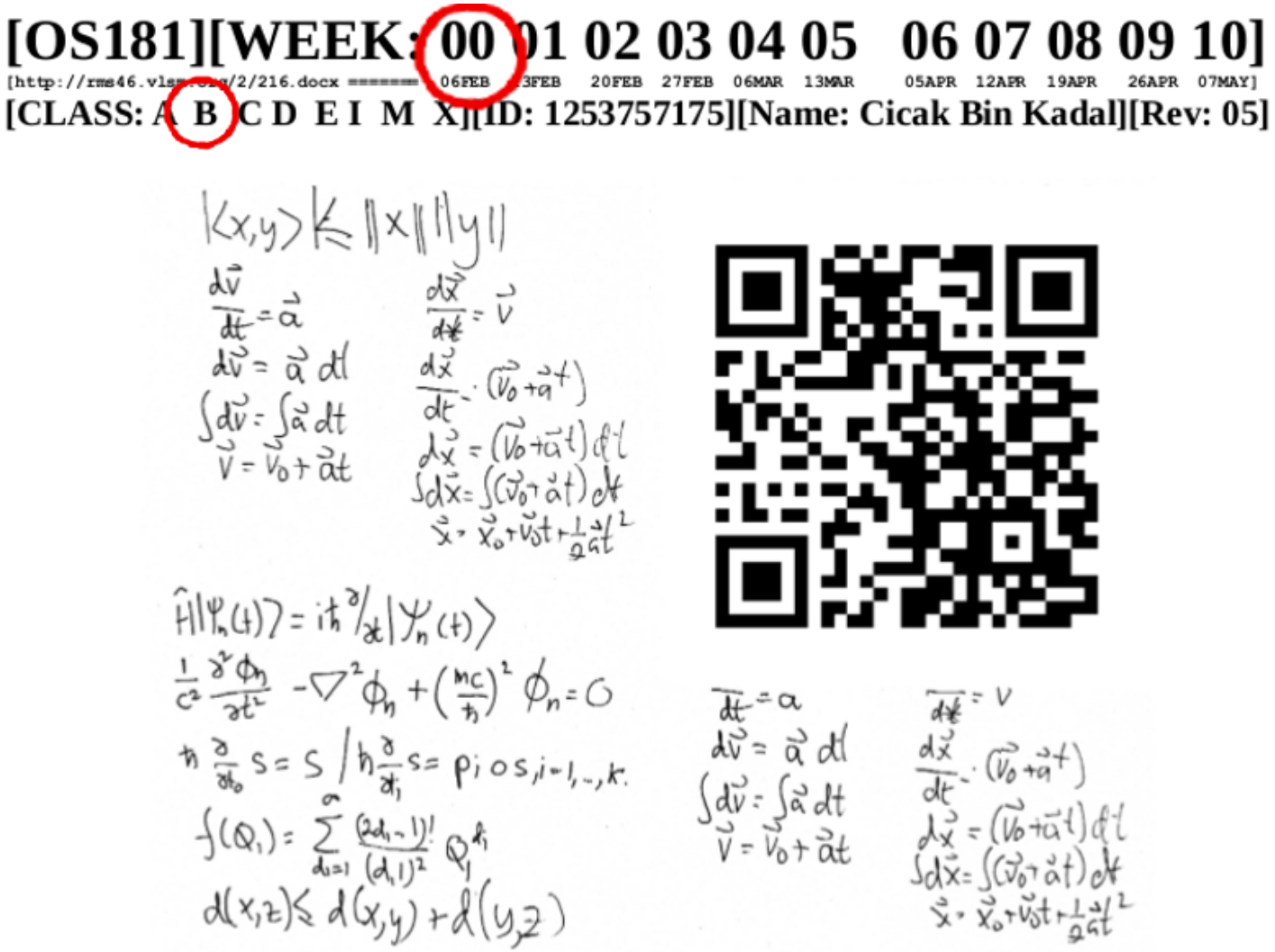


Figure: **Memo**: OS181 B 1253757175 cicak.git cicak.sso Cicak Bin Kadal

ETC

- Week 00: Send QRC to `os181@vlsm.org`
  - Subject: [W00] CLASS ID SIAK-NAME
  - Example:
    - Mailto: os181@vlsm.org
    - Subject: [W00] B 125375 Cicak Bin Kadal
    - Insert your QR Code (embedded).
- Masalah Administratip
  - Harap menghubungi SEKRE (Ged. B lt. 2) untuk segala masalah administratip, terutama absen, sakit, surat sakit, ujian susulan, dst.
  - Harap merampungkan masalah administrasi ujian susulan dalam 6 hari kerja.

Arsip Scele

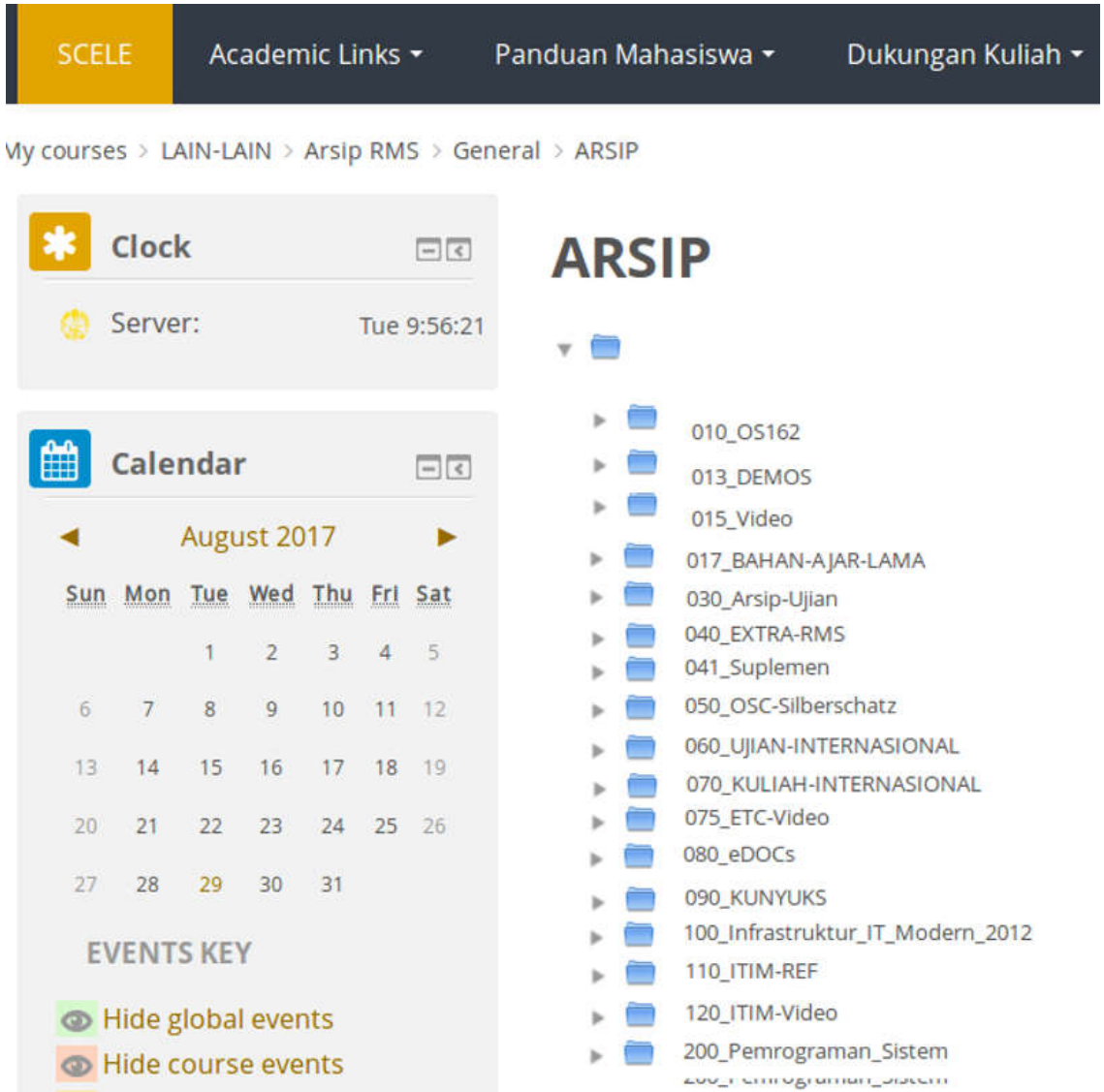


Figure: Lihat juga BADAK.cs.ui.ac.id:/extra/

Bahan presentasi: http://rms46.vlsm.org/2/207.html

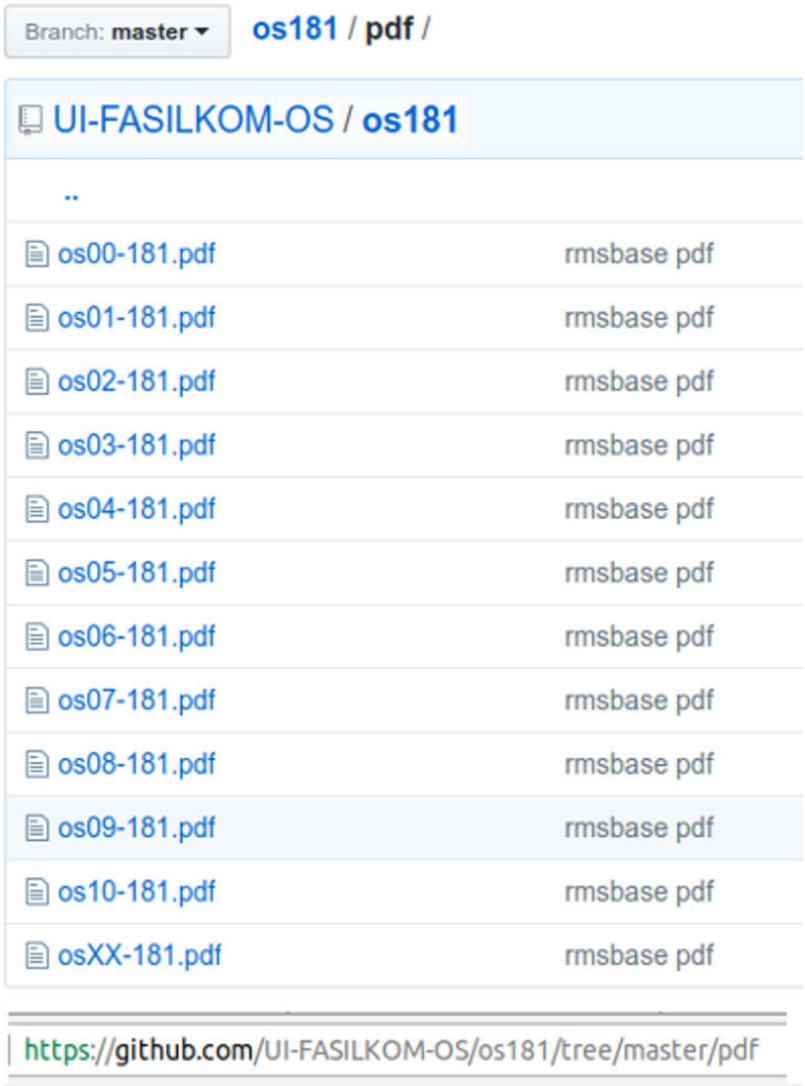


Figure: Alternatif: BADAK.cs.ui.ac.id:/extra/os181/pdf/

BADAK.cs.ui.ac.id:/extra/

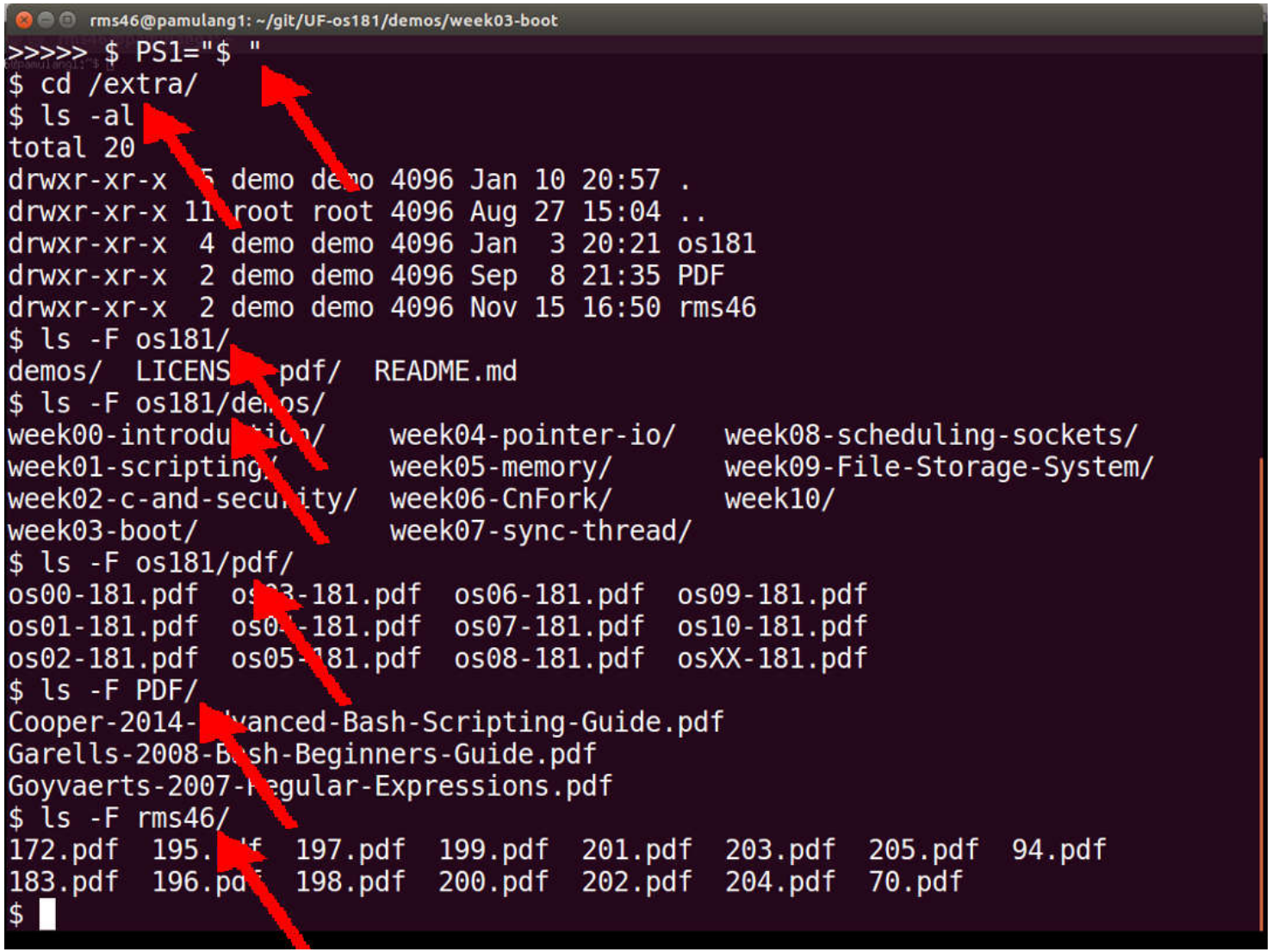


Figure: BADAK.cs.ui.ac.id:/extra/

WSL 1

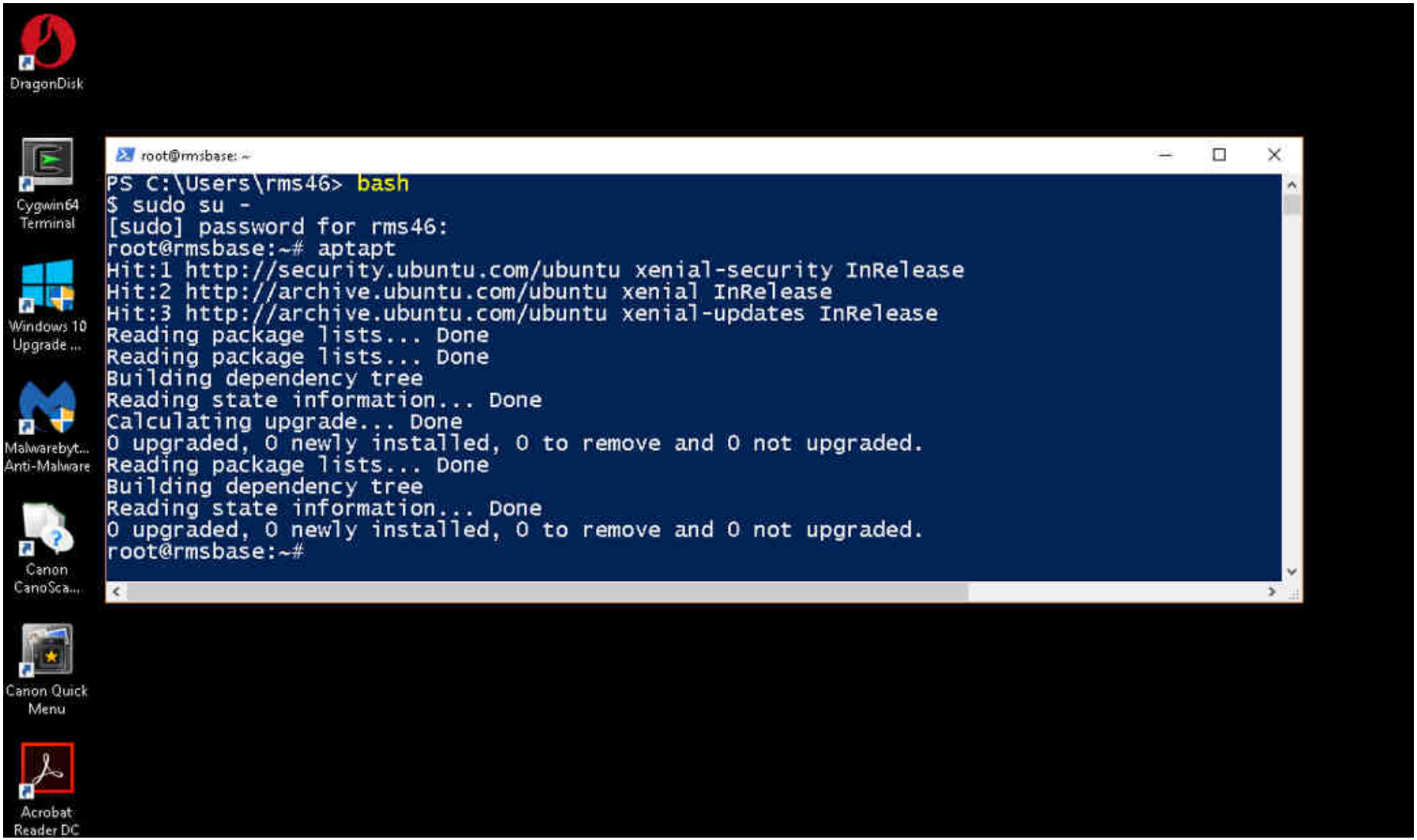


Figure: WSL: Windows Subsystem for Linux

WSL 2

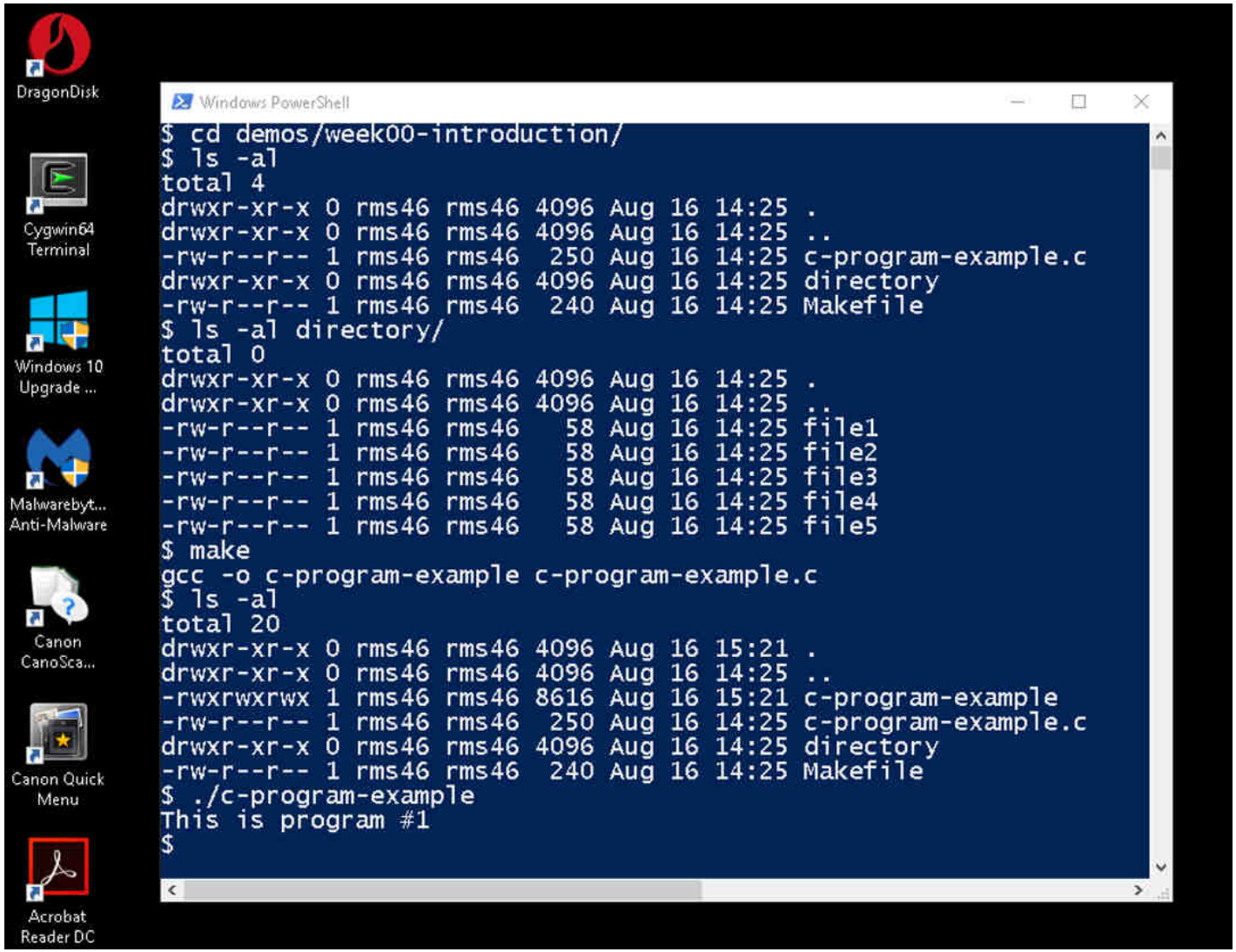


Figure: WSL: Windows Subsystem for Linux

Cygwin

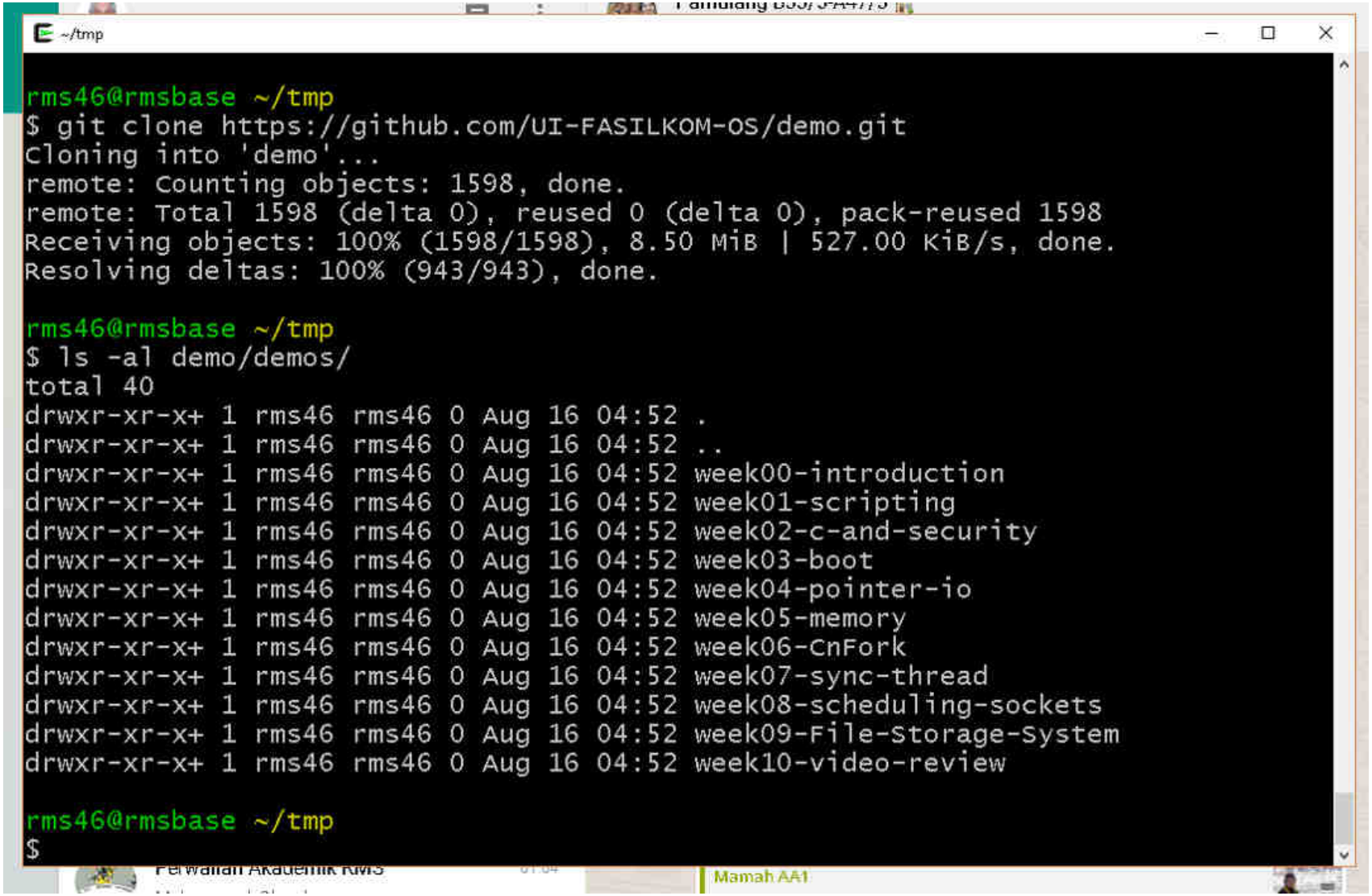


Figure: Cygwin

Cygwin

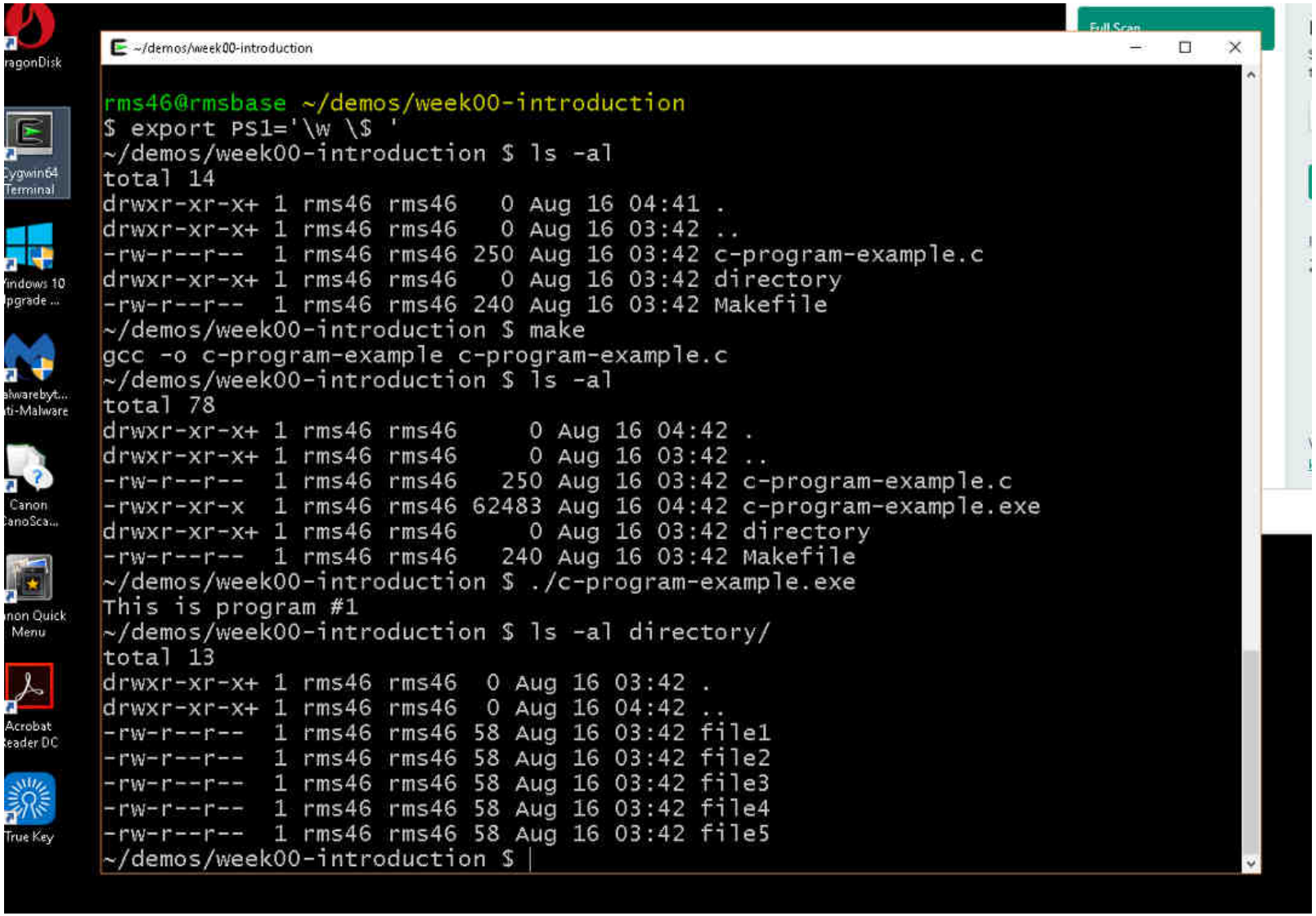


Figure: Cygwin

Github Demo

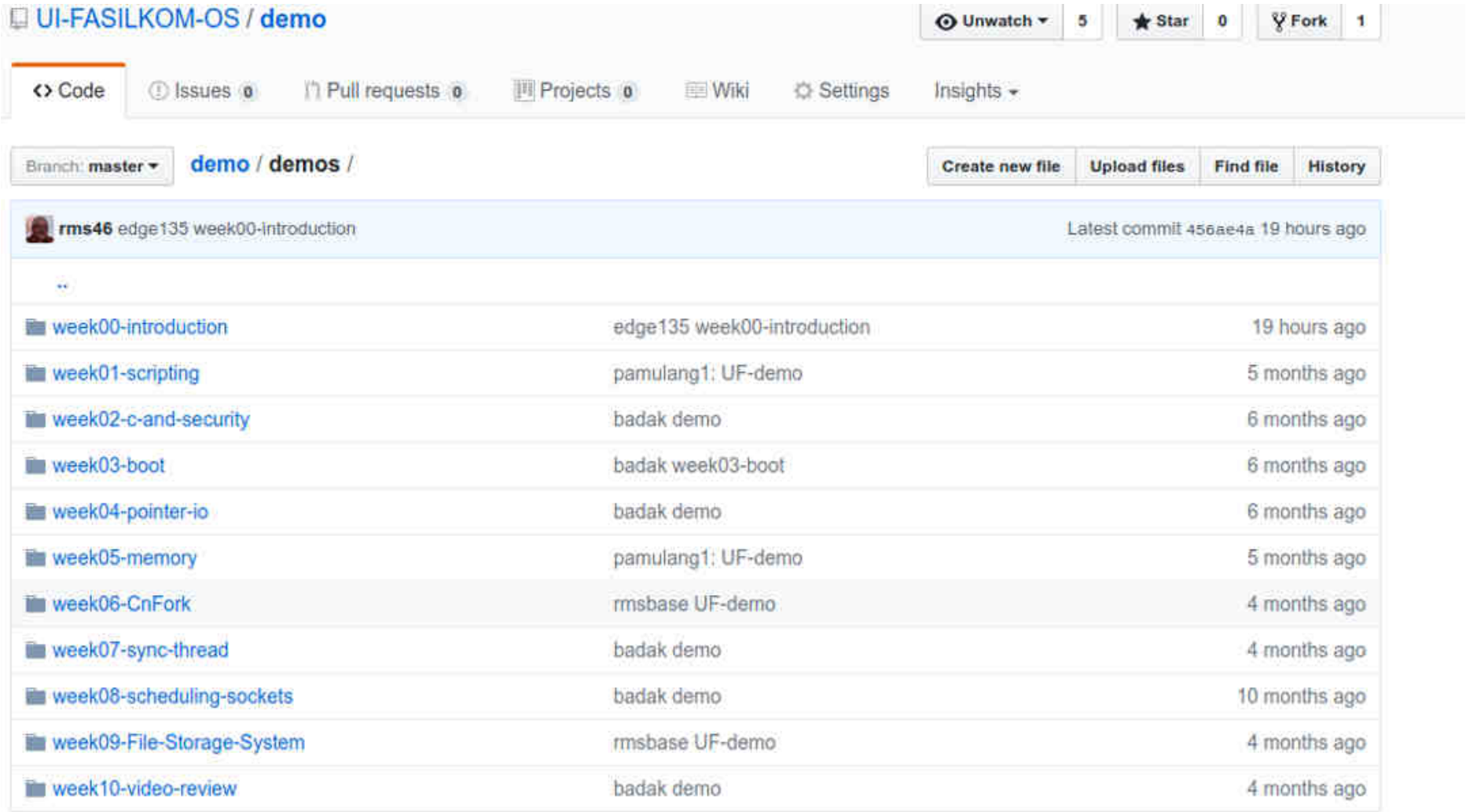


Figure: `https://github.com/UI-FASILKOM-OS/demo`

Login: Kawung dan Badak

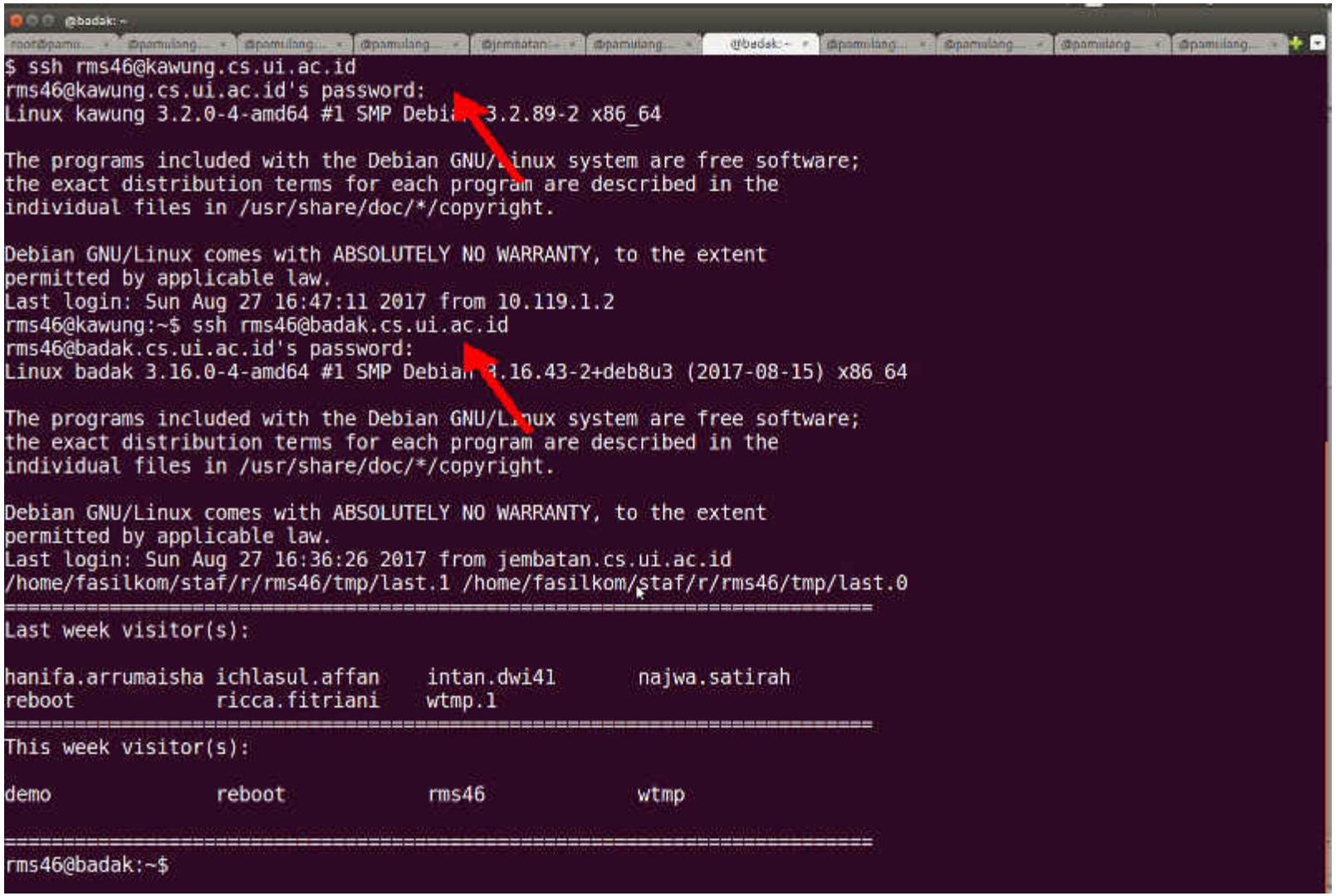


Figure: Login: Kawung dan Badak

Program Example

```
$ cat c-program-example.c
/* (c) 2016-2017 Rahmat M. Samik-Ibrhaim
 *  REV01 Sun Aug 20 15:01:12 WIB 2017
 *  START Fri Jan 01 00:00:00 WIB 2016
 *  This is a free software.
 *  To compile:
 *  $ gcc -o c-program-example c-program-example.c
 *  To execute:
 *  $ ./c-program-example
 */

#include <stdio.h>

void main() {
   printf("This is program #1\n");
}
```

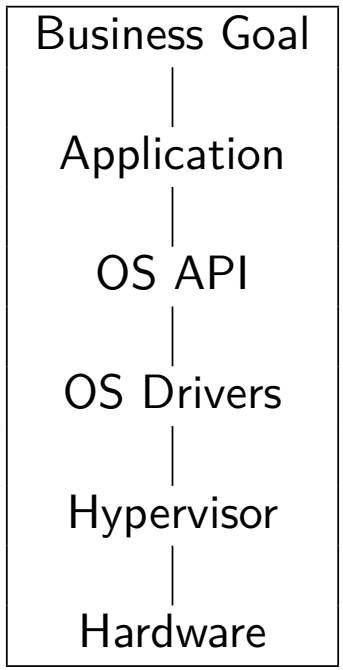
Makefile

```
$ cat Makefile

# (c) 2016-2017 Rahmat M. Samik-Ibrahim
# REV01 Tue Aug 22 14:45:14 WIB 2017
# START Fri Jan 01 00:00:00 WIB 2016
# This is a free Makefile configuration.
# Just run:
# % make

ALL:  c-program-example

c-program-example: c-program-example.c
   gcc -o c-program-example c-program-example.c

clean:
   rm -f c-program-example
```

```
$ ls -al
total 44
drwxr-xr-x  3 rms46 rms46  4096 Aug 28 18:45 .
drwxr-xr-x 13 rms46 rms46  4096 Feb 28 18:50 ..
-rw-r--r--  1 rms46 rms46   334 Aug 23 20:17 c-program-example.c
drwxr-xr-x  2 rms46 rms46  4096 Sep 26  2016 directory
-rw-r--r--  1 rms46 rms46   319 Aug 23 20:17 Makefile
-rw-r--r--  1 rms46 rms46 23606 Aug 28 18:26 QuickResponseCode
                                                        .docx
$ make
gcc -o c-program-example c-program-example.c
$ ./c-program-example
This is program #1
$ ls -al
total 56
........
$ make clean
rm -f c-program-example
$
```

Week 00: Introduction

- Reference: (OSCE2e ch1/2)(UCB 01)(UDA P1L1/2)(OLD 00)
- Operating System
  - Why take this OS class?
  - Definition: Resource Allocator & Control Program.
  - Managers: Process, Memory, Storage, ...
  - Layers
  - Interfaces



Computer Organization Review

- You should understand:
  - von Neumann Model.
  - Buses, Bridges, Transfer Rate, Clock.
  - Memory: DDR, DDR-2, ...
  - Cache, Buffer, Spool.
  - Direct Memory Access (DMA).
  - Port & Memory Mapped I/O.
  - CPU: privilege/kernel/supervisor mode and user mode.
  - Hardware Limitation.
  - Priority: Read vs Write.
  - Interrupts: Polling & Vectored.
  - Multiprocessors: Symmetric vs. Asymmetric.
  - Multicore & Multithreading.
  - Clustered Systems.
  - Numbers: base 2, base 8, base 10, base 16.
    - Base 2: 110010101010_2
    - Base 8: $01234567_8 = 000\ 001\ 010\ 011\ 100\ 101\ 110\ 111_2$
    - Base 10: 012 345 679
    - Base 16: $9AB\ CDEF_{16} = 1001\ 1010\ 1011\ \ 1100\ 1101\ 1110\ 1111_2$

Block Diagram

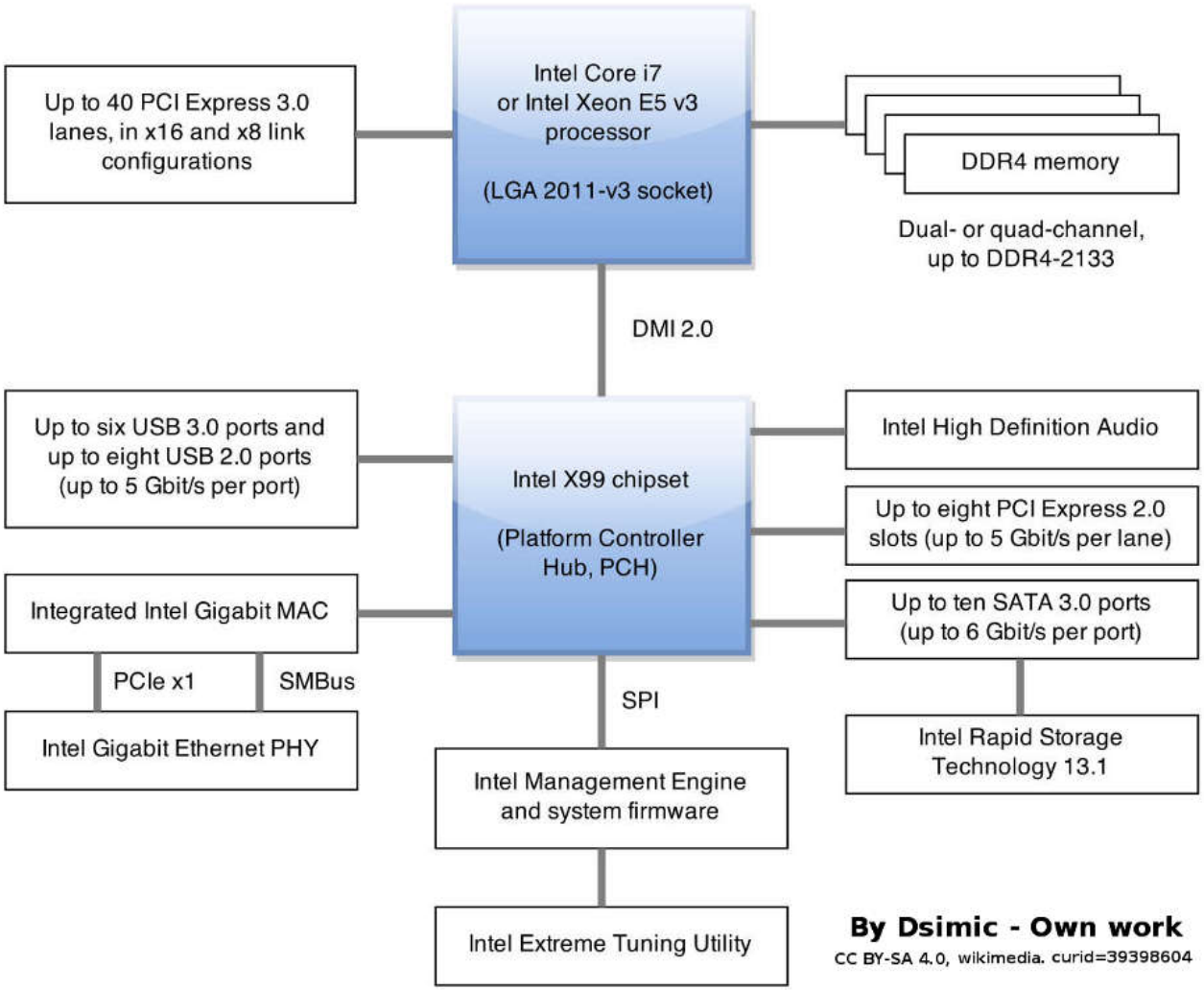


Figure: Block Diagram

APIC

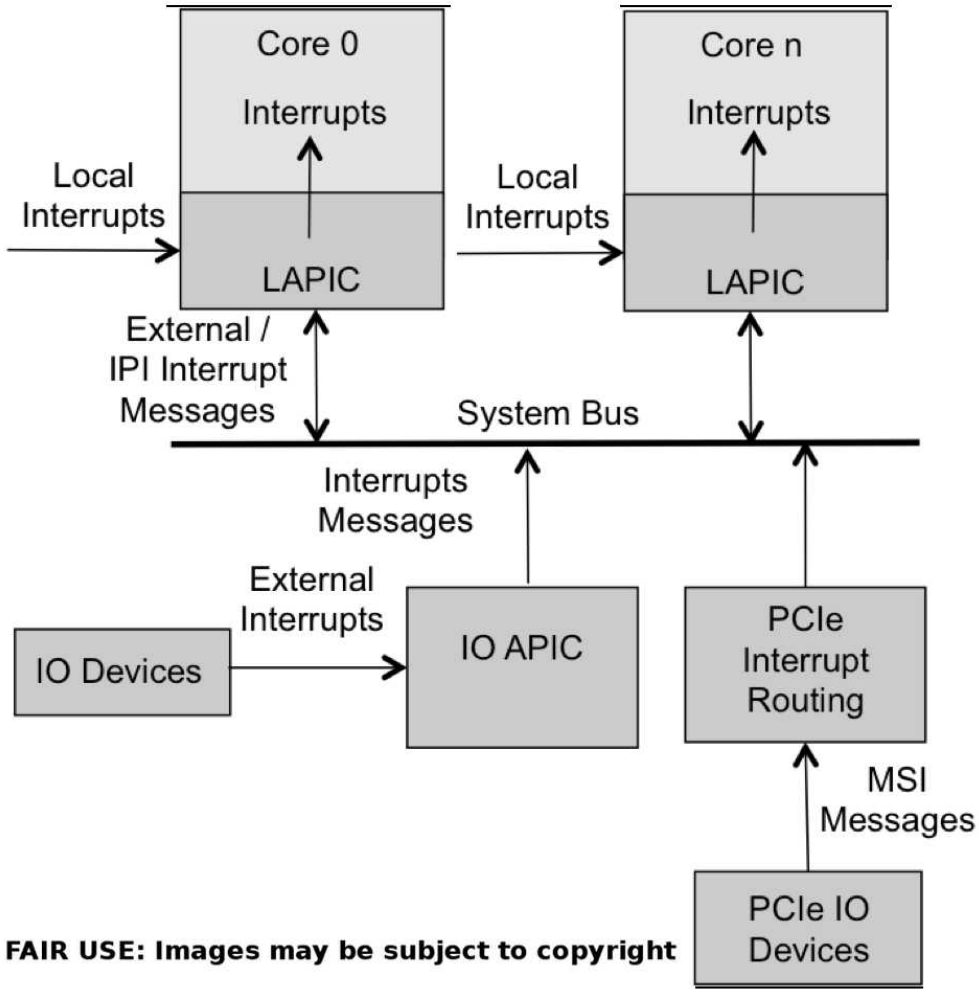


Figure: APIC (Advanced Programmable Interrupt Controller)

Interupt Handling

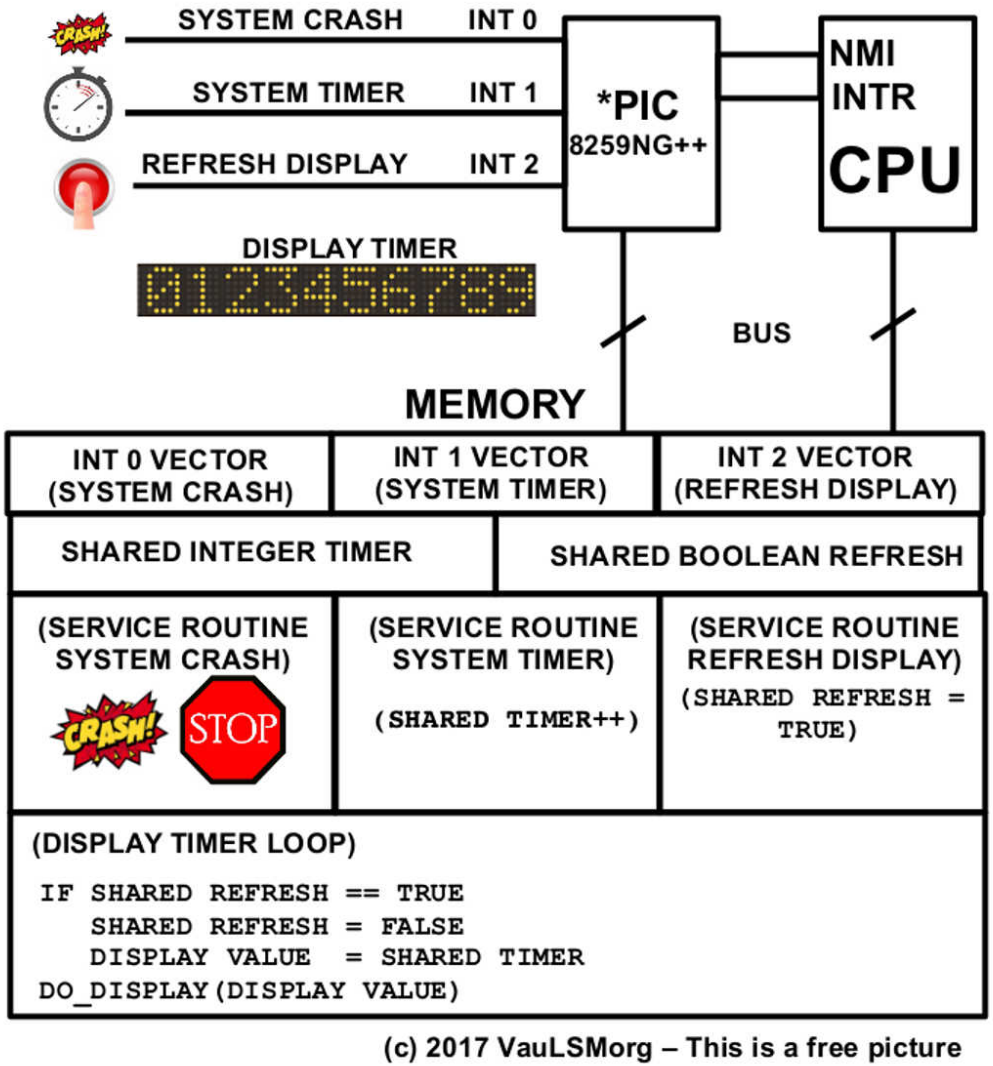


Figure: Interupt Handling with PIC (Programmable Interrupt Controller)

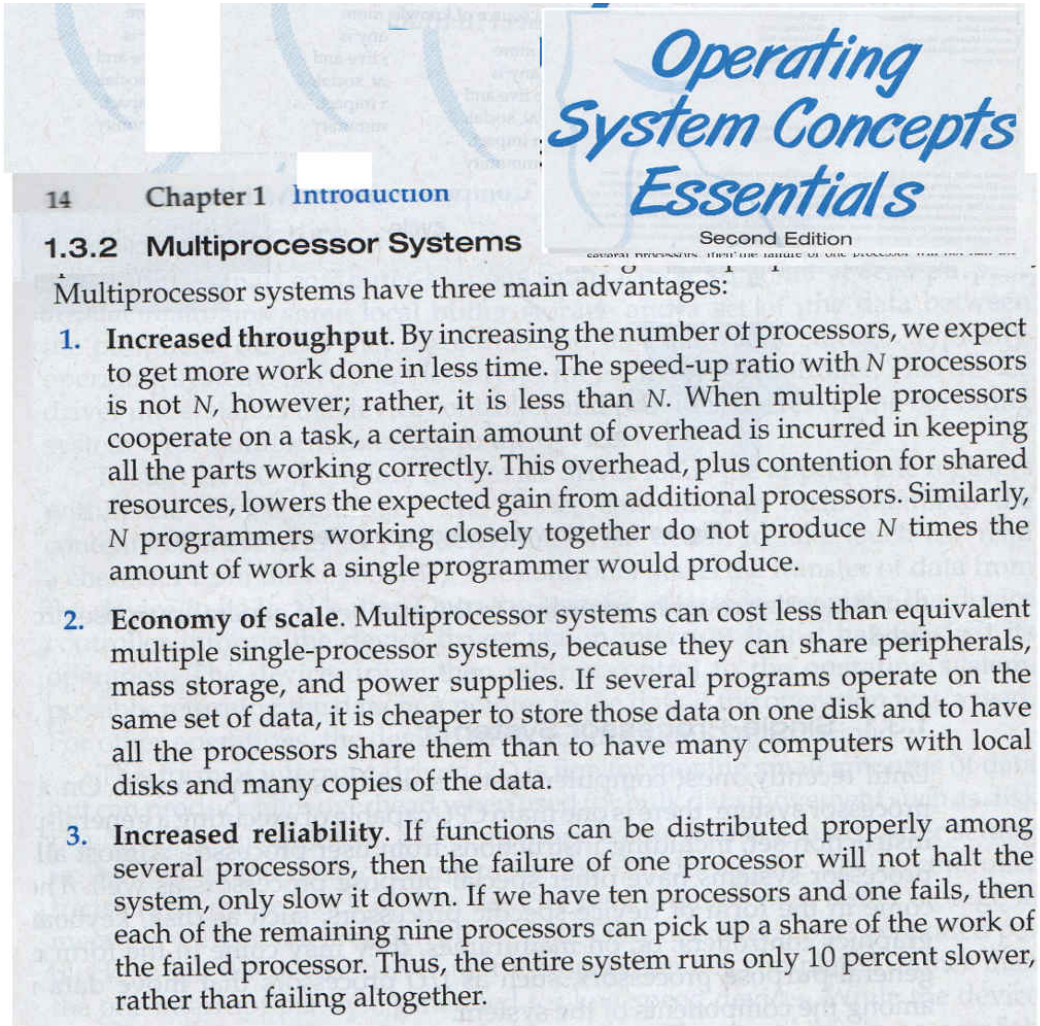
Managers Set

- Process:
  - Creating/Deleting; Suspending/Resuming; Synchronization; Communication; Schedulling
- Memory:
  - Tracking; Move In/Move Out; Allocating/Deallocating.
- Storage/File System:
  - Create/Delete; Open/Close; Read/Write.
- Mass Storage:
  - Schedulling; Allocating; Free Space.
- I/O:
  - Buffering; Caching; Spooling.
  - Interfacing (driving).
- Protecting & Security:
  - Protecting.
  - Security.

Potpourri

- Mobile/Distributed/Client-Server/Peer-to-Peer Computing.
- Real-Time Computing: Hard Real-Time vs. Soft Real-Time.
- Operating System Comparison: Android, *BSD, GNU/Linux, iOS, Mac OS, Windows.
- Operating System Services: UI (GUI, CLI); Program Executing; I/O Operations; File Systems Manipulation; Communication; Error Detection; Resource Allocation; Accounting; Protection & Security.
- System Calls: Process Control; File Management; Device Management; Information Maintenance; Communications; Protection.
- Application Programming Interface (API)
- Standard C Library.
- System Programs.
- Microkernel System Structure.
- Loadable Kernel Modules.
- Virtualization and Cloud System.

Aneka Soal Ujian Sistem Operasi Week00 2016-1 (OSC2e)



Operating System Concepts Essentials
Second Edition

14 Chapter 1 Introduction

1.3.2 Multiprocessor Systems

Multiprocessor systems have three main advantages:

1. **Increased throughput**. By increasing the number of processors, we expect to get more work done in less time. The speed-up ratio with N processors is not N, however; rather, it is less than N. When multiple processors cooperate on a task, a certain amount of overhead is incurred in keeping all the parts working correctly. This overhead, plus contention for shared resources, lowers the expected gain from additional processors. Similarly, N programmers working closely together do not produce N times the amount of work a single programmer would produce.
2. **Economy of scale**. Multiprocessor systems can cost less than equivalent multiple single-processor systems, because they can share peripherals, mass storage, and power supplies. If several programs operate on the same set of data, it is cheaper to store those data on one disk and to have all the processors share them than to have many computers with local disks and many copies of the data.
3. **Increased reliability**. If functions can be distributed properly among several processors, then the failure of one processor will not halt the system, only slow it down. If we have ten processors and one fails, then each of the remaining nine processors can pick up a share of the work of the failed processor. Thus, the entire system runs only 10 percent slower, rather than failing altogether.

Figure: **T / F** The advantages of a multiprocessor system include: increased throughput, economy of scale, and increased reliability.

Week 00 Check List

- ☐ Creat project "os181" on your new (or existing) github.com account.
- ☐ Encode the QR Code.
- ☐ Mailto: os181@vlsm.org (Subject: [W00] CLASS ID SIAK-NAME)

The End

- ☐ This is the end of the presentation.
- ☒ This is the end of the presentation.
- This is the end of the presentation.