CSGE602055 Operating Systems
CSF2600505 Sistem Operasi
Week 01: Overview 2, Virtualization & Scripting

Rahmat M. Samik-Ibrahim (ed.)

University of Indonesia

https://os.vlsm.org/Slides/os01.pdf
Always check for the latest revision!

REV321 26-Jul-2021

Operating Systems 212[3]) — **PJJ from HOME** ZOOM: A [Xxx XX:XX] — B [Xxx XX:XX] — INT [Xxx XX:XX]

Week	**Schedule & Deadline**[1])	**Topic**	**OSC10**[2])
Week 00	XX Xxx - XX Xxx 2021	Overview 1, Virtualization & Scripting	Ch. 1, 2, 18.
Week 01	XX Xxx - XX Xxx 2021	Overview 2, Virtualization & Scripting	Ch. 1, 2, 18.
Week 02	XX Xxx - XX Xxx 2021	Security, Protection, Privacy, & C-language.	Ch. 16, 17.
Week 03	XX Xxx - XX Xxx 2021	File System & FUSE	Ch. 13, 14, 15.
Week 04	XX Xxx - XX Xxx 2021	Addressing, Shared Lib, & Pointer	Ch. 9.
Week 05	XX Xxx - XX Xxx 2021	Virtual Memory	Ch. 10.
Week 06	XX Xxx - XX Xxx 2021	Concurrency: Processes & Threads	Ch. 3, 4.
Week 07	XX Xxx - XX Xxx 2021	Synchronization & Deadlock	Ch. 6, 7, 8.
Week 08	XX Xxx - XX Xxx 2021	Scheduling + W06/W07	Ch. 5.
Week 09	XX Xxx - XX Xxx 2021	Storage, Firmware, Bootloader, & Systemd	Ch. 11.
Week 10	XX Xxx - XX Xxx 2021	I/O & Programming	Ch. 12.

[1]) The **DEADLINE** of Week 00 is XX Xxx 2021, whereas the **DEADLINE** of Week 01 is XX Xxx 2021, and so on...

[2]) Silberschatz et. al.: **Operating System Concepts**, 10^{th} Edition, 2018.

[3]) This information will be on **EVERY** page two (2) of this course material.

STARTING POINT — https://os.vlsm.org/

- **Text Book** — Any recent/decent OS book. Eg. (**OSC10**) Silberschatz et. al.: **Operating System Concepts**, 10^{th} Edition, 2018. See also https://www.os-book.com/OS10/.
- **Resources**
  - **SCELE OS212** — https://scele.cs.ui.ac.id/course/view.php?id=XXXX. The enrollment key is **XXX**.
  - **Download Slides and Demos from GitHub.com** https://github.com/UI-FASILKOM-OS/SistemOperasi/: os00.pdf (W00), os01.pdf (W01), os02.pdf (W02), os03.pdf (W03), os04.pdf (W04), os05.pdf (W05), os06.pdf (W06), os07.pdf (W07), os08.pdf (W08), os09.pdf (W09), os10.pdf (W10).
  - **Problems** — https://rms46.vlsm.org/2/: 195.pdf (W00), 196.pdf (W01), 197.pdf (W02), 198.pdf (W03), 199.pdf (W04), 200.pdf (W05), 201.pdf (W06), 202.pdf (W07), 203.pdf (W08), 204.pdf (W09), 205.pdf (W10).
  - **LFS** — http://www.linuxfromscratch.org/lfs/view/stable/
  - **OSP4DISS** — https://osp4diss.vlsm.org/
  - **DOIT** — https://doit.vlsm.org/001.html

Agenda

Agenda (2)

Week 01 Overview II: Topics[1]

- Types of virtualization (including Hardware/Software, OS, Server, Service, Network)
- Paging and virtual memory
- Virtual file systems
- Hypervisors
- Portable and cost of virtualization; emulation vs. isolation
- Cloud services: IAAS, PAAS and Platform APIs, SAAS
- Introduction to Scripting and REGEX.

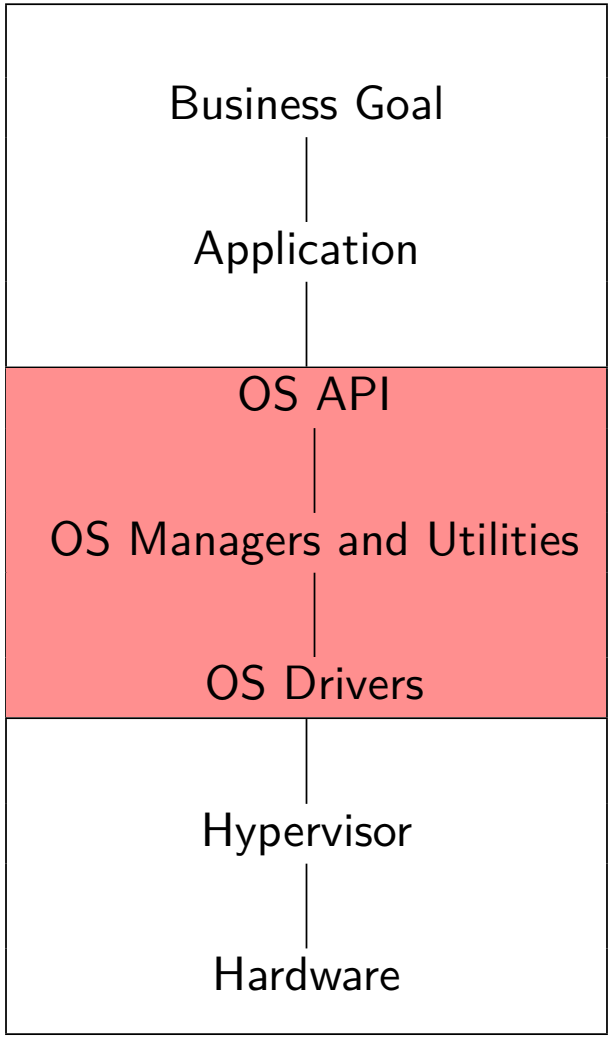
[1] Source: ACM IEEE CS Curricula 2013

Week 01 Overview II: Learning Outcomes[1]

- Explain the concept of virtual memory and how it is realized in hardware and software. [Familiarity]
- Discuss hypervisors and the need for them in conjunction with different types of hypervisors. [Usage]
- Differentiate emulation and isolation. [Familiarity]
- Evaluate virtualization trade-offs. [Assessment]
- Discuss the importance of elasticity and resource management in cloud computing. [Familiarity]
- Explain the advantages and disadvantages of using virtualized infrastructure. [Familiarity]

[1] Source: ACM IEEE CS Curricula 2013

The Operating System



Week 01: Review 2 & Scripting

- Pengenalan Lisensi Perangkat Lunak Bebas: https://rms46.vlsm.org/1/70.pdf
- Intellectual Property Right (IPR)
- Operating System Services
- User Operating System Interface
- System Calls
- Types of System Calls
- System Programs
- Operating System Design and Implementation
- Operating System Structure

Intelectual Property Right (IPR)

- Trade Secret (Rahasia Dagang) — UU no. 30/2000.
- Industrial Design (Desain Industri) — UU no. 31/2000.
- Integrated Circuit Layout Design (Desain Tata Letak Sirkuit Terpadu) — UU no. 32/2000.
- Paten *(Patent)* — UU no. 14/2001.
- Copyright (Hak Cipta) — UU no. 19/2002.
- The problem of Intellectual Property Right (IPR).
- Software IPR.
- Software Licenses: GNU GPL, EULA. Public Domain, Apache, Microsoft Public License.

Is this a Software Patent or Not?

The AV1 Video Codec — Timothy B. Terriberry — LINUX.CONF.AU 21-25 January 2019

EUROPEAN PATENT SPECIFICATION

(11) EP 0 460 751 B1

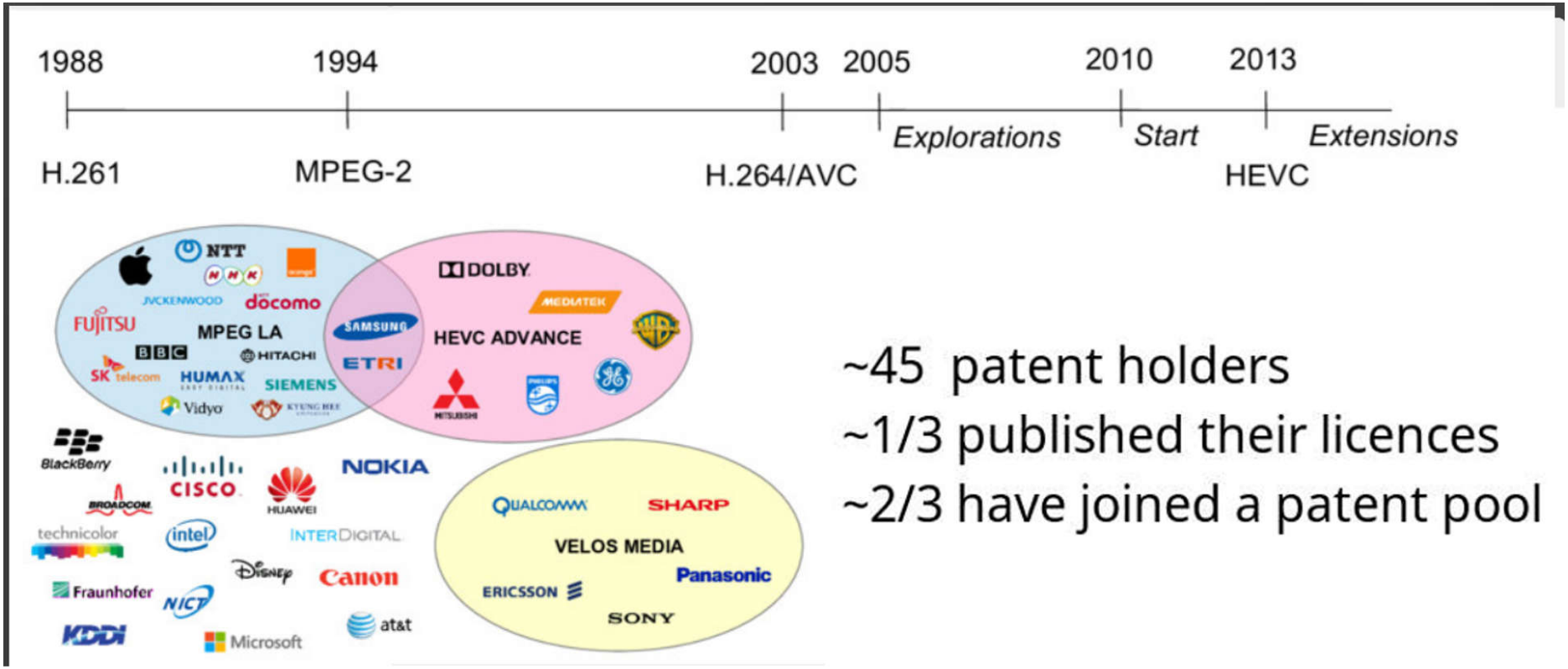
Date of filing: **03.06.1991**

Method of transmitting audio and/or video signals

Description

The invention relates to a method of transmitting audio and/or video signals *via* some transmission medium. More particularly the transmission medium is constituted by an optically readable disc. However, the transmission medium may also be a magnetic tape or disc or a direct connection between a transmitter and a receiver. The invention also relates to the transmission medium on which the audio and/or video signals are recorded, to an encoding apparatus for transmitting the audio and/or video signals, and to a decoding apparatus for receiving these signals.

The Codec Mess



~45 patent holders
~1/3 published their licences
~2/3 have joined a patent pool

Courtesy of
Jonatan Samuelsson
Divideon
Co-founder and CEO

Alliance for Open Media

Source (per 21-Sep-2020): https://aomedia.org/membership/members/

Free Software

- Free Software Definition (FSF)
  0. The freedom to run the program as you wish, for any purpose (freedom 0).
  1. The freedom to study how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
  2. The freedom to redistribute copies so you can help your neighbor (freedom 2).
  3. The freedom to distribute copies of your modified versions to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.
- Free Software vs. Open Source Software.
- Copyleft Software.

Software Licenses

- 3-clause BSD license and 2-clause BSD license (BSD-X-Clause)
- Apache License 2.0 (Apache-2.0)
- Artistic License 2.0 (ArtisticLicense2)
- Common Development and Distribution License (CDDL-1.0)
- Eclipse Public License (EPL-1.0)
- Educational Community License 2.0 (ECL2.0)
- Expat License (Expat) aka. MIT license (MIT)
- GNU Affero General Public License v3 (AGPL-3.0)
- GNU All-Permissive License (GNUAllPermissive)
- GNU General Public License (GPL)
- GNU Lesser General Public License (LGPL)
- Microsoft Public License (MS-PL)
- Mozilla Public License 2.0 (MPL-2.0)
- "Public Domain" (PublicDomain)
- X11 License (X11License)

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.

Virtualization & Cloud Computing

- Virtual Machine
  - Host & Guest
  - Hypervisor (Virtual Machine Manager)
    - Type 0, 1, 2 Hypervisor
    - ParaVirtualization
    - Programming-environment Virtualization
    - Emulators
  - Application Containment (OS-Level)
    - Containers: LXC, Solaris Containers, Docker.
    - Zones: Solaris Containers
    - Virtual Private Servers: OpenVZ
    - Virtual Kernels: DragonFly BSD
    - Jails: FreeBSD Jail/ Chroot Jail
  - Kubernetes (K8s): A (open source) system for managing CONTAINERIZED applications.
- Cloud Computing
  - SAAS: Software As A Service.
  - PAAS: Platform As A Service.
  - IAAS: Infrastructure As A Service.

Week 01: Assignment #1: VirtualBox Guest Preparation

- Visit https://osp4diss.vlsm.org/#idx00b.
- You need a hypervisor (VirtualBox), which can be installed on Windows 10, macOS, or Linux.
  - Downloading and Installing VirtualBox
    https://www.virtualbox.org/
- Using a Download Manager might help for a less reliable internet connection.
  - FDM: Free Download Manager (Optional)
    https://www.freedownloadmanager.org/
- For Windows 10, you also need to install PUTTY and WINSCP.
  - Visit https://osp4diss.vlsm.org/SSHGuest.html
- Understand how to Import / Export / Delete a Virtual Guest
  - Visit https://osp4diss.vlsm.org/#idx01.
  - Exporting OVA
  - Importing OVA
  - Export / Import OVA
  - Deleting OVA

Week 01: Assignment #2: Importing or Installing?

- **Option 1 or Option 2?**
  - Option 1: You might **import** a pre-installed OVA file.
    - README: https://bit.ly/3B12SU6 (371 bytes)
    - Debian 10 OVA for VirtualBox: https://bit.ly/36z3diC (346MB)
    - See also https://osp4diss.vlsm.org/#idx01.
  - Option 2: You might **install** a Debian Guest from scratch.
    - Visit https://osp4diss.vlsm.org/#idx02b. Set an **EMPTY VIRTUAL GUEST**, General, System, Display, Storage, Audio, Network.
    - Download Debian Netinst (ISO).
    - Installing Debian NetInst (guest).
- Running a VirtualBox Debian Guest (OVA or ISO)
  - Visit https://osp4diss.vlsm.org/#idx03.
  - Try to startup, console login, console log out, and shutdown.
  - Try ssh (or putty), and scp (or winscp).
  - Study some Command Lines, Editor, Regular Expression, and String Processing. (See also assignment #3).
  - Shutdown and export your virtual guest (Guest-01a.ova).

Week 01: Assignment #3 The ATM Way: GSGS and Read

- Do the **ATM Way**[1]: **GSGS**[2] and Read!
  - **REVISIT** https://osp4diss.vlsm.org/Welcome2GNULinux.html
    - Linux Sucks 2021 https://youtu.be/WtJ9T_IJOPE?t=87
    - Basic Linux Commands https://youtu.be/CpTfQ-q6MPU and https://linoxide.com/linux-command/essential-linux-basic-commands/
    - The Complete Linux Course: Beginner to Power User (7:23 hours) – https://youtu.be/wBp0Rb-ZJak
    - Basic vi commands https://youtu.be/ggSyF1SVFr4 and https://www.cs.colostate.edu/helpdocs/vi.html.
    - Learn REGEX in 15 minutes https://youtu.be/bgBWp9EIlMM
    - BASH https://tldp.org/LDP/abs/abs-guide.pdf, https://youtu.be/F-gskSl4pwQ, https://youtu.be/_n5ZegzieSQ.
    - SED https://www.gnu.org/software/sed/manual/sed.pdf
    - GAWK https://www.gnu.org/software/gawk/manual/gawk.pdf
    - Essential Awk https://youtu.be/9YOZmI-zWok
    - Linux Man Pages https://youtu.be/uJnrh9hAQR0
    - The Minix3 Notes: https://rms46.vlsm.org/2/166.pdf
  - **VISIT** https://osp4diss.vlsm.org/osp-115.html

[1] Amati, Tiru, Modifikasi. Romi Satria Wahono has been using this term since 2007.

[2] Google Sana, Google Sini

Week 01: Assignment #3 (1): Learn/Try It

- Revisit https://osp4diss.vlsm.org/
- Learn some basic **Command-Line Interface** (CLI) commands: bash, cat, cd, cp, ls, man, more, mv, rm, touch, wc, vi, sed, awk, git.
- Learn how to **turn on** and **turn off** (shutdown) your Virtual Guest.
- Learn how to **login** and **logout** with `ssh` or `putty`.
- Learn some basic Regular Expression (regex).
- Learn the concept of a Stream Editor (sed).
- Learn the concept of a Git, GitHub, and GitHub Pages.
- Learn the concept of a AWK.
- Learn the concept of a SCRIPTING.
- Pick and learn how to use an editor, eg. (**vi**).

Week 01: Assignment #3 (2): Some Essential Commands

man	manual. Eg. "man man"
passwd	changes passwords.
ls	list directory contents. Eg. "ls -al"
cd	change the working directory. Eg. "cd /tmp"
cp	copy file(s). Eg. "cp SOURCE DEST"
rm	remove file(s). Eg. "rm AFILE"
mv	move files(s). Eg. "mv FROMFILE TOFILE"
mkdir	make directories(s). Eg. "mkdir ADIRECTORY"
rmdir	remove directories(s). Eg. "rmdir ADIRECTORY"
cat	read file(s) Eg. "cat AFILE"
more	read file(s) per screen Eg. "more AFILE"
ln	make a link of a file. Eg. "ln -s file sfile"
grep	search string aword inside file. Eg. "grep aworld file"
sort	sort lines of text files. Eg. "sort file1.txt"
top	display systems task. Eg. "top"
find	Eg. "find / -name minix3.iso -print". Find from "/".

Week 01: Assignment #3 (3): Some Essential Commands

chmod	Eg. `"chmod 755 file"`. Change file with access mode 755.
chown	Eg. `"chown user file"`. Change owner file to user.
chgrp	Eg. `"chgrp other file"`. Change group file to other.
tar	tape archive file. Eg.
	`"tar cf /tmp/tfile.tar dir/"`. Archive "dir/" into tfile.tar.
	`"tar tf /tmp/tfile.tar"`. List tfile.tar.
	`"tar xf /tmp/tfile.tar"`. Extract tfile.tar.
date	print or set the system date and time. Eg. `"date +%Y"`
tee	read from standard input and write to standard output and files.
	Eg. `"ls -al | tee listing.txt"`
diff	compare files line by line. Eg. `"diff file1.txt file2.txt"`
wc	print newline, word, and byte counts for each file.
	Eg. `"wc file.txt"`

Week 01: Assignment #3 (4): The "vi" editor

- VI Basics

	Basics		More Commands
i	insert mode	d^	delete from ^ (beginning) to the curs
a	append mode	d$	delete from the cursor to $ (end)
<ESC>	escape mode	dd	delete the whole line
q!	quit	5dd	delete 5 lines
wq!	write and quit	yy	yank (copy) the line
ZZ	write and quit	p	put (paste) the line
h j k l	move [left, down, up, right]	J	joint current and next line
r	replace a character	:r file.txt	read (insert) file.txt
d	delete a character	:w! file.txt	write into file.txt
u	undo	:1,8 w! file.txt	write line 1 to 8 into file.txt

  - Basic vi Commands
    https://www.cs.colostate.edu/helpdocs/vi.html
  - Vim Basics in 8 Minutes https://youtu.be/ggSyF1SVFr4

Week 01: Assignment #4: Dress Up Your Virtual Guest

- See https://osp4diss.vlsm.org/#idx04.
- Rename Guest as `Guest-01b.ova`
- Update Your Debian Guest
- Adding Debian Packages
- Your Virtual Guest Account should be == Your GitHub Account. E.g. "`cbkadal`". See Adding User Name
- Your Virtual Guest Account should be == Your Virtual Hostname. E.g. "`cbkadal`". See Rename a Hostname
- Create/Edit ".bash_profile"
- Create/Edit "vimrc"
- Create/Edit ".bash_aliases"
- SUPERUSER
- Export Guest as `Guest-01b.ova`

Week 01: Assignment #5: SSH Keys for Git

- See https://osp4diss.vlsm.org/#idx05.
- Rename Guest as `Guest-01c.ova`
- SSH: Generating public/private rsa key pair
- SSH: Put a Public Key at GitHub.com (e.g. cbkadal)
- GIT: .gitconfig
- GIT: cloning from GitHub.com

Week 01: Assignment #6: Dress Up Your GitHub Page

- GSGS about the Markdown (MD) markup language.
  - See https://github.com/adam-p/markdown-here/wiki/Markdown-Cheatsheet.
  - See https://doit.vlsm.org/003.html
  - See https://template.vlsm.org/. You may want to download https://template.vlsm.org/template.zip
- Create a markdown file "links.md".
  - Fill the page with links that (you think) are useful for this Operating Systems course.
  - Add two sentences:
    - what the link about is
    - why you think the link is interesting.
  - The links will be peer reviewed.
- For example, see cbkadal's:
  - GitHub Page at https://cbkadal.github.io/os212/LINKS/.
  - https://raw.githubusercontent.com/cbkadal/os212/master/links.md.
- Your links may appear next semester in https://osp4diss.vlsm.org/osp-115.html.

Week 01: Assignment #7: Update mylog.txt

- mylog: Updating add commit push
  - Using your favorite editor (perhaps "vi"?), update file mylog.txt.
- See `https://osp4diss.vlsm.org/ETC/logCodes.txt` for the Code list.
- **PUSH BACK** your repo into GitHub.com!
- Always remember the **4 Git Mantras**:

```
git pull
git add -A
git commit
git push
```

- Export Guest as `Guest-01c.ova`

Week 01: Assignment #8 (1): More awk, bash, regex, sed

- awk
  - `awk '{print "Hello awk!"}' file.txt` — print "Hello awk!" for every file.txt line.
  - `awk '{print $0}' file.txt` — print every file.txt line.
  - `awk '{print $1}' file.txt` — print first field of every file.txt line.
  - `awk '{print $2}' file.txt` — print second field of every file.txt line.
- regex
  - to search patterns
  - BRE (Basic Regular Expression) vs ERE (Extended Regular Expression)
  - Flavors: Grep, Java, JavaScript, PHP, POSIX, Python, sed, XML, ...

Week 01: Assignment #8 (2): bash, regex, sed, and awk

- ≪`^$`≫ — matches a beginning-of-line + end-of-line (empty line).
  - ≪`^`≫ — matches a beginning-of-line (meaningless).
  - ≪`^hello$`≫ — matches just "hello" in a line.
- ≪`.`≫ — matches any character.
  - ≪`hell.`≫ — matches "hellA", "hella", "hellB", "hellb", ...
- ≪`[AB]`≫ — matches "A" or "B" only.
  - ≪`[0-3]`≫ — matches "0", "1", "2", or "3" only.
  - ≪`[^4-9]`≫ — not match "4", "5", "6", "7", "8", or "9".
- ≪`?`≫ — matches preceding zero or one time.
  - ≪`a?b`≫ — matches "b" or "ab" only.
- ≪`*`≫ — matches preceding zero or more times.
  - ≪`a*b`≫ — matches "b" or "ab" or "aab" or ...
  - ≪`A.*Z`≫ — matches "AZ" or "AaZ" or "AabZ" or ...
- ≪`+`≫ — matches preceding one or more times.
  - ≪`a+b`≫ — matches "ab", "aab", "aaab", ...
- ≪`{}`≫ — matches numbers in {}.
  - ≪`a{2}`≫ — matches "aa".
  - ≪`a{2,5}`≫ — matches "aa", "aaa", "aaaa", and "aaaaa".
  - ≪`a{2,}`≫ — matches "aa", "aaa", "aaaa", "aaaaa", ...

Week 01: Assignment #8 (3): bash, regex, sed, and awk

- ≪\≫ — escape character.
- ≪\0≫ — NULL.
- ≪\b≫ — word boundary.
- ≪\B≫ — non-word boundary.
- ≪\d≫ — any digit. Eg. ≪\d{1,3}≫ = 0 - 999.
- ≪\D≫ — any non-digit.
- ≪\n≫ — new line.
- ≪\t≫ — tab.
- ≪\s≫ — white space character.
- ≪\S≫ — non white space character.

Week 01: Assignment #8 (4): bash, regex, sed, and awk

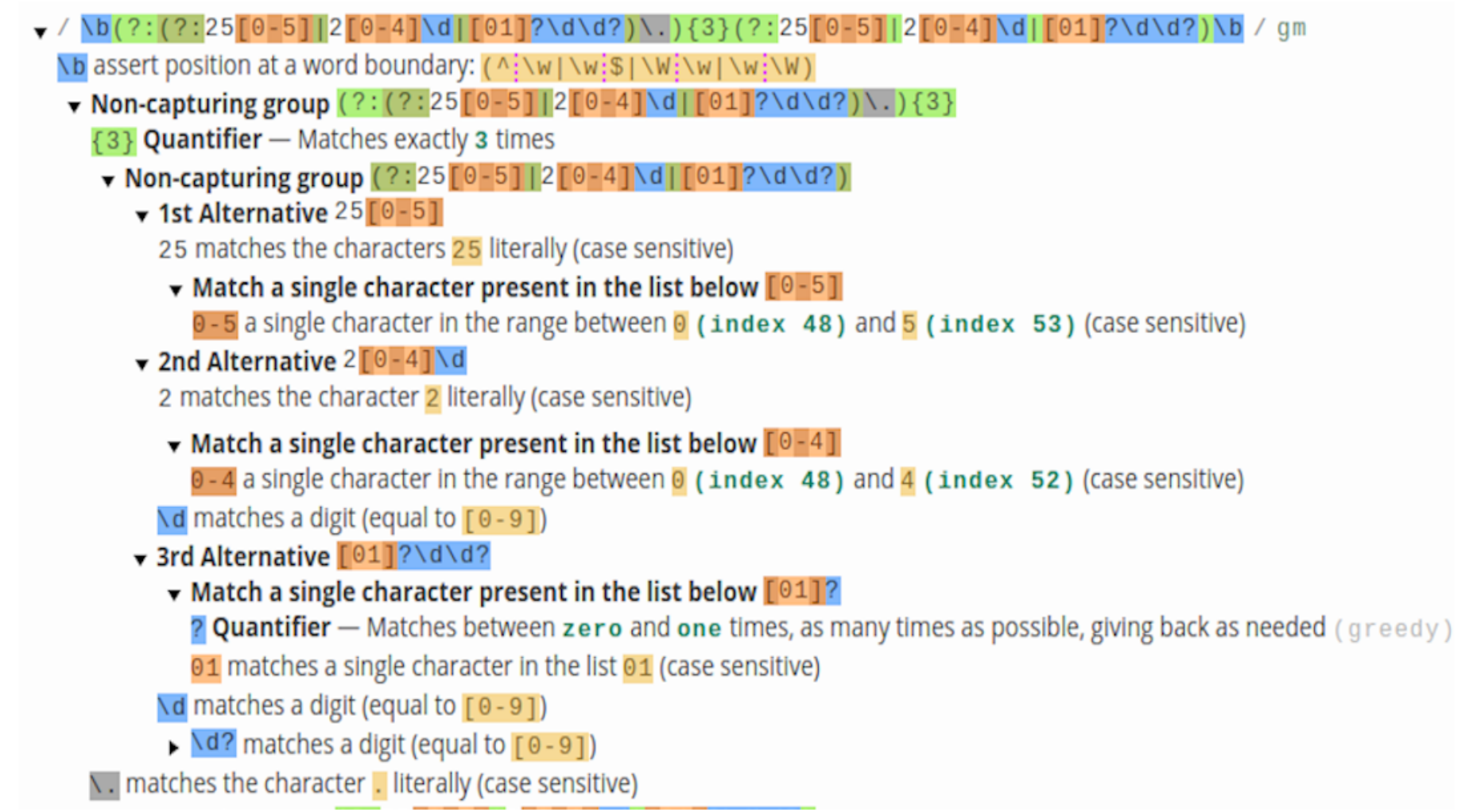
- ≪`(...)`≫ — group.
  - ≪`(?:...)`≫ — passive group.
  - ≪`(regex)|(regex)`≫ — matches left regex or right regex.
  - ≪`(a|b`≫ — matches either a or b.
  - ≪`^(From|To):`≫ — matches either ≪`^From:`≫ or ≪`^To:`≫.
- ≪`[0-9]{10}`≫ — 10 digits.
- ≪`0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]`≫ — 00:00–23:59.
- ≪`([0-9]|0[0-9]|1[0-9]|2[0-3]):[0-5][0-9]`≫ — (0)0:00–23:59.

Week 01: Assignment #8 (5): bash, regex, sed, and awk

- ≪`[:alnum:]`≫ — alpha-numerics.
- ≪`[:alpha:]`≫ — alphabets
- ≪`[:blank:]`≫ — spaces and tabs.
- ≪`[:digit:]`≫ — digits.
- ≪`[:lower:]`≫ — lower case.
- ≪`[:space:]`≫ — spaces.
- ≪`[:upper:]`≫ — upper case.
- ≪`[:xdigit:]`≫ — hexadecimal digits.
- ≪`[:punct:]`≫ — punctuation.
- ≪`[:cntrl:]`≫ — control characters.
- ≪`[:graph:]`≫ — printed characters.
- ≪`[:print:]`≫ — printed and spaces.
- ≪`[:word:]`≫ — alpha-numerics and underscore.

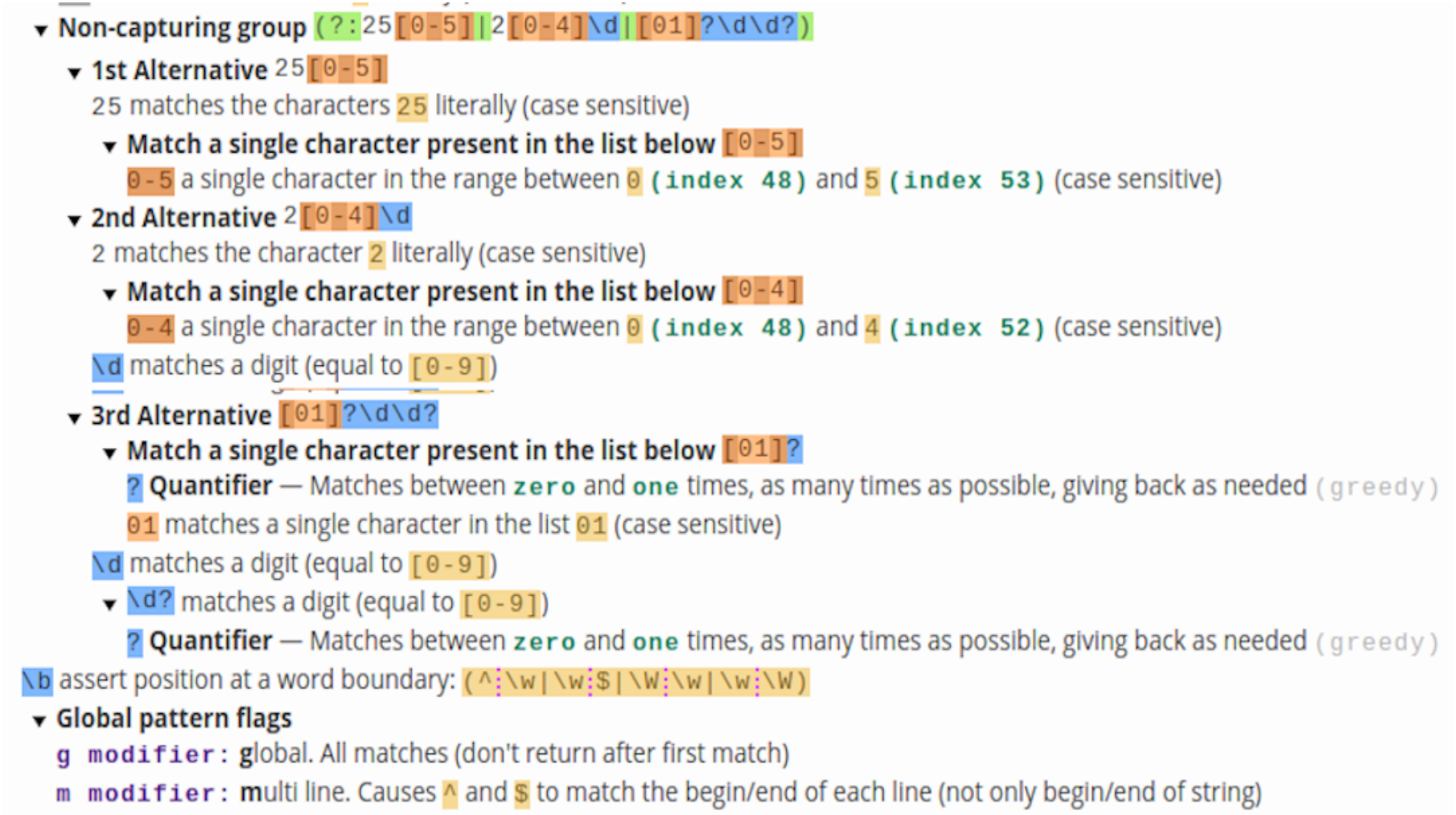
Week 01: Assignment #8 (6): bash, regex, sed, and awk

```
\b(?:(?:25[0-5]|2[0-4]\d|[01]?\d\d?)\.){3}
(?:25[0-5]|2[0-4]\d|[01]?\d\d?)\b
```



Week 01: Assignment #8 (8): bash, regex, sed, and awk

```
\b(?:(?:25[0-5]|2[0-4]\d|[01]?\d\d?)\.){3}
(?:25[0-5]|2[0-4]\d|[01]?\d\d?)\b
```



Week 01: Assignment #8 (9): bash, regex, sed, and awk

```
# file.txt

1. This is no. 1.
2. This is no. 22.
3. This is no. 333.
4. This is no. 4 4 4 4.
5. This is Joko.
6. This is Joko Joko.
7. This is joko.
8. This is Bowo.
9. This is bowo.
sed    'G'    ZA-thisfile1.txt
sed    'G;G'  ZA-thisfile1.txt
sed -n '4,6p' ZA-thisfile1.txt
sed -n '4,6p' ZA-thisfile1.txt > ZA-thisfile2.txt
sed -n '/[0-9]\{2\}/p' ZA-thisfile1.txt
sed    '4,6d' ZA-thisfile1.txt
sed    '$d' ZA-thisfile1.txt
sed    '5,/HABATS/d'   ZA-thisfile1.txt
sed    's/Joko/Bowo/'   ZA-thisfile1.txt
sed    's/Joko/Bowo/2' ZA-thisfile1.txt
sed    's/Joko/Bowo/g' ZA-thisfile1.txt
sed    's/Bowo\|bowo/Joko/g' ZA-thisfile1.txt
awk    '{print "Hello awk!"}' ZA-thisfile1.txt
awk    '{print $0}' ZA-thisfile1.txt
awk    '{print $1}' ZA-thisfile1.txt
awk    '{print $2}' ZA-thisfile1.txt
HABATS: This is the last line, dude!
```

Week 01: Assignment #8 (10): bash, regex, sed, and awk

- `sed 'G' file.txt` — double space.
- `sed 'G;G' file.txt` — triple space.
- `sed -n '4,6p' file.txt` — show only line 4 to 6.
- `sed -n '4,6p' file.txt > newfile.txt` — write line 4 to 6 to newfile.txt.
- `sed '/[0-9]\{2\}/p' file.txt` — show only lines with two digits.
- `sed '4,6d' file.txt` — show all except line 4 to 6.
- `sed '$d' file.txt` — show all except last line.
- `sed '5,/HABATS/d'` — show all except from line 5 to a line with HABATS.
- `sed 's/Joko/Bowo/' file.txt` — replace Joko with Bowo.
- `sed 's/Joko/Bowo/2' file.txt` — replace the second Joko with Bowo.
- `sed 's/Joko/Bowo/g' file.txt` — replace every Joko with Bowo.
- `sed 's/Bowo\|bowo/Joko/g' file.txt` – replace every Bowo or bowo with Joko.

Week 01: Check List (Deadline: tba).

- This page is https://os.vlsm.org/Slides/check01.pdf.
- **Starting Point:** https://os.vlsm.org/
- Week 01: Assignment (more details in **os01.pdf**).
  1. VirtualBox Guest Preparation
  2. Importing (OVA) or Installing (ISO)?
  3. Do the ATM way: GSGS and Read!
  4. Dress Up Your Virtual Guest
  5. SSH Keys for Git
  6. Dress Up Your GitHub Page
  7. Update mylog.txt
  8. More awk, bash, regex, sed

The End

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