# CSGE602055 Operating Systems
# CSF2600505 Sistem Operasi
# Week 08: Scheduling

Rahmat M. Samik-Ibrahim (ed.)

University of Indonesia

`https://os.vlsm.org/`
Always check for the latest revision!

REV187 04-Feb-2019

# Operating Systems 2019-1

A (Rm 3114) [Tu/Th 10-12] — B (Rm 3114) [Tu/Th 13-15] — C (Rm 3114)
[Tu/Th 16-18] — D (Rm 2401) [Tu/Th 10-12] — E (Rm 2306) [Tu/Th 13-15]

| Week | Schedule | Topic | OSC10 |
|------|----------|-------|-------|
| Week 00 | 07 Feb - 13 Feb 2019 | Overview 1, Virtualization & Scripting | Ch. 1, 2, 18. |
| Week 01 | 14 Feb - 20 Feb 2019 | Overview 2, Virtualization & Scripting | Ch. 1, 2, 18. |
| Week 02 | 21 Feb - 27 Feb 2019 | Security, Protection, Privacy, & C-language | Ch. 16, 17 |
| Week 03 | 28 Feb - 06 Mar 2019 | File System & FUSE | Ch. 13, 14, 15 |
| Week 04 | 12 Mar - 18 Mar 2019 | Addressing, Shared Lib, & Pointer | Ch. 9 |
| Week 05 | 19 Mar - 25 Mar 2019 | Virtual Memory | Ch. 10 |
| Mid-Term | 23-30 Mar 2019 (tba) | MidTerm (UTS) | |
| Week 06 | 02 Apr - 08 Apr 2019 | Concurency: Processes & Threads | Ch. 3, 4 |
| Week 07 | 09 Apr - 15 Apr 2019 | Synchronization & Deadlock | Ch. 6, 7, 8 |
| Week 08 | 16 Apr - 22 Apr 2019 | Scheduling | Ch. 5 |
| Week 09 | 23 Apr - 29 Apr 2019 | Storage, BIOS, Loader, & Systemd | Ch. 11 |
| Week 10 | 30 Apr - 06 May 2019 | I/O & Programming | Ch. 12 |
| Reserved | 07 May - 17 May 2019 | | |
| Final | 18-25 May 2019 (tba) | Final (UAS) | This schedule is |
| Extra | 27 Jun 2019 | Extra assignment confirmation | subject to change. |

## The Weekly Check List

- ☐ **Resources**: `https://os.vlsm.org/`
    - ☐ **(THIS) Slides** — `https://github.com/UI-FASILKOM-OS/SistemOperasi/tree/master/pdf/`
    - ☐ **Demos** — `https://github.com/UI-FASILKOM-OS/SistemOperasi/tree/master/demos/`
    - ☐ **Extra** — `BADAK.cs.ui.ac.id:///extra/`
    - ☐ **Problems** — `https://rms46.vlsm.org/2/`:
      `195.pdf` (Week 00), `196.pdf` (Week 01), `197.pdf` (Week 02),
      `198.pdf` (Week 03), `199.pdf` (Week 04), `200.pdf` (Week 05),
      `201.pdf` (Week 06), `202.pdf` (Week 07), `203.pdf` (Week 08),
      `204.pdf` (Week 09), `205.pdf` (Week 10).
- ☐ **Text Book**: any recent/decent OS book. Eg. (**OSC10**) Silberschatz et. al.: **Operating System Concepts**, $10^{th}$ Edition, 2018.
- ☐ Encode your **QRC** with size upto 7cm x 7cm (ca. 400x400 pixels):
  **"OS182 CLASS ID SSO-ACCOUNT Your-Full-Name"**
- ☐ Write your Memo (with QRC) **every week**.
- ☐ Login to `badak.cs.ui.ac.id` via `kawung.cs.ui.ac.id` for at least **10 minutes** every week. Copy the weekly demo files to your own home directory.
  Eg. (Week00): `cp -r /extra/Week00/W00-demos/ W00-demos/`

# Agenda

# Week 08 Scheduling: Topics[1]

- Preemptive and non-preemptive scheduling
- Schedulers and policies
- Processes and threads
- Deadlines and real-time issues

---

[1]Source: ACM IEEE CS Curricula 2013

# Week 08 Scheduling: Learning Outcomes[1]

- Compare and contrast the common algorithms used for both preemptive and non-preemptive scheduling of tasks in operating systems, such as priority, performance comparison, and fair-share schemes. [Usage]
- Describe relationships between scheduling algorithms and application domains. [Familiarity]
- Discuss the types of processor scheduling such as short-term, medium-term, long-term, and I/O. [Familiarity]
- Describe the difference between processes and threads. [Usage]
- Compare and contrast static and dynamic approaches to real-time scheduling. [Usage]
- Discuss the need for preemption and deadline scheduling. [Familiarity]
- Identify ways that the logic embodied in scheduling algorithms are applicable to other domains, such as disk I/O, network scheduling, project scheduling, and problems beyond computing. [Usage]

[1]Source: ACM IEEE CS Curricula 2013

# Week 08: Scheduling

- Reference: (OSC10-ch05 demo-w08)
- Scheduling
  - Basic Concepts
    - **WARNING:** It's just a BURST
    - IO Burst
    - CPU Burst
    - CPU Burst vs. Freq (See next slide)
  - Criteria: Utilization, throughput, {turnaround, waiting, response} time.
  - (Burst) Algorithm
    - FCFS, SJF, RR, Priority, Multilevel Queue.
  - Preemptive / Non-preemptive (Cooperative) Scheduling
  - I/O Bound / CPU Bound Processes
- Thread Scheduling
  - User-level $\rightarrow$ Process-Contention Scope (PCS): many to many/one.
  - Kernel-level $\rightarrow$ System-Contention Scope (SCS): one to one.
- Standard Linux Scheduling
  - Completely Fair Scheduler (CFS).
  - Real Time Scheduling.

# CPU Burst: How Long (When)?



©2013 Silberschatz, Galvin and Gagne   Operating System Concepts – 9th Edition

# MultiProcessor Schedulling

- Asymmetric Multiprocessing vs. Symmetric Multiprocessing (SMP).
- Processor Affinity: soft vs. hard.
- NUMA: Non-Uniform Memory Access.
- Load Balancing
- Multicore Processors
- Real Time Schedulling: Soft vs. Hard.
- Big O Notation
  - O(1)
  - O(log N)
  - O(N)

# The Two State Model

- CPU State – I/O State – CPU State – ...
  - n: processes in memory.
  - p: I/O time fraction.
  - $p^n$: probability n processes waiting for I/O.
  - $1 - p^n$: CPU utilization of n processes.
  - $\left[\frac{(1-p^n)}{n}\right]$: CPU utilization of ONE processes.
- Example: $p = 60\% \Rightarrow$ **CPU Utilization Per Process**: $\left[\frac{1-(60\%)^n}{n}\right]$

| CPU Utilization | Multiprogramming (%) | | | | |
|---|---|---|---|---|---|
| **N** | **1** | **2** | **3** | **4** | **5** |
| **Per Process** | 40 | 32 | 26 | 21 | 18 |

- For 5 concurrent processes:
  If total time is 100 seconds; for each processs, the CPU time will be 18 seconds.

# The End

☐ This is the end of the presentation.

☒ This is the end of the presentation.

- This is the end of the presentation.