

CSGE602055 Operating Systems
CSF2600505 Sistem Operasi
Week 08: Scheduling + W06/W07

C. BinKadal

Sendirian Berhad

<https://doc0S.vlsm.org/Slides/os08.pdf>

Always check for the latest revision!

REV418: Tue 30a Jan 2024 22:00

OS241³): Operating Systems Schedule 2023 - 2

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

¹⁾ For schedule, see <https://os.vlsm.org/#idx02>

²⁾ Silberschatz et. al.: **Operating System Concepts**, 10th 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**, 10th Edition, 2018. (See <https://codex.cs.yale.edu/avi/os-book/OS10/>).
- Resources** (<https://os.vlsm.org/#idx03>)
 - SCELE** — <https://scele.cs.ui.ac.id/course/view.php?id=3743>.
The enrollment key is **XXX**.
 - Download Slides and Demos from GitHub.com** —
(<https://github.com/os2xx/docOS/>)
[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**
[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/>
 - This is How Me Do It!** — <https://doit.vlsm.org/>
 - PS: "Me" rhymes better than "I", duh!

Agenda

- 1 Start
- 2 OS241 Schedule
- 3 Agenda
- 4 Week 08
- 5 OSC10 (Silberschatz) Chapter 5
- 6 Scheduling
- 7 CPU Burst: How Long (When)?
- 8 MultiProcessor Scheduling
- 9 The Two State Model

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

¹Source: ACM IEEE CS Curricula

Week 08 Scheduling: Learning Outcomes¹

- 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]

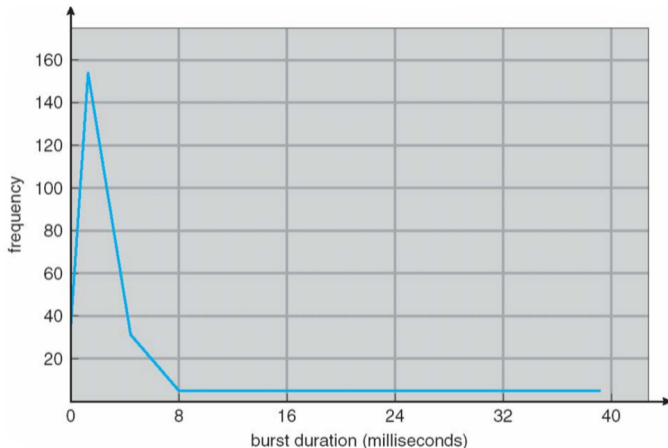
¹Source: ACM IEEE CS Curricula

- OSC10 Chapter 5: CPU Scheduling
 - Basic Concepts
 - Scheduling Criteria
 - Scheduling Algorithms
 - Thread Scheduling
 - Multi-Processor Scheduling
 - Real-Time CPU Scheduling
 - Operating Systems Examples
 - Algorithm Evaluation

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 → Process-Contention Scope (PCS): many to many/one.
 - Kernel-level → 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

Figure: Burst: Duration vs Frequency

MultiProcessor Scheduling

- Asymmetric Multiprocessing vs. Symmetric Multiprocessing (SMP).
- Processor Affinity: soft vs. hard.
- NUMA: Non-Uniform Memory Access.
- Load Balancing
- Multicore Processors
- Real Time Scheduling: 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 (%)				
	1	2	3	4	5
Per Process	40	32	26	21	18

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