

# Sistem Operasi 2009



## Pertemuan 2

# Sistem Operasi

(Pengenalan)

H u s n i

Lab. Sistem Komputer & Jaringan  
Teknik Informatika Univ. Trunojoyo

# Ikhtisar

---

- Definisi Sistem Operasi
- Evolusi Sistem Operasi
- Pencapaian Besar
- Sistem Operasi Modern
- Review Sistem Windows, Unix & Linux

# Sistem Operasi

---

- Suatu program yang mengontrol eksekusi dari program aplikasi
- Suatu antarmuka (*interface*) antara aplikasi dan hardware
- Obyektif: Convenience, Efficiency, Ability to evolve.

# Layer & View

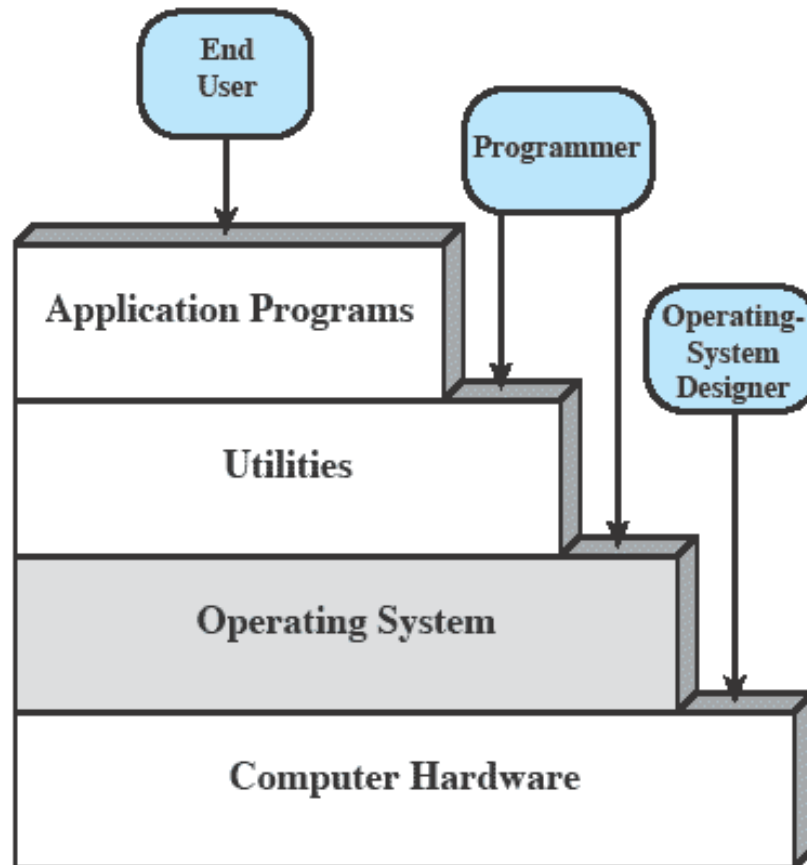


Figure 2.1 Layers and Views of a Computer System

# Layanan dari SO (1)

---

- Pengembangan program
- Eksekusi program
- Akses terhadap perangkat I/O
- Akses terkendali terhadap file
- Akses terkontrol terhadap sistem (public system)

# Layanan dari SO (2)

---

- Deteksi dan Respon Terhadap Error
  - Error hardware Internal & Eksternal
  - Error software
  - SO tidak dapat mengijinkan permintaan aplikasi

# Layanan dari SO (3)

---

- Accounting
  - Menghimpun statistika penggunaan
  - Memonitor kinerja
  - Digunakan untuk mengantisipasi peningkatan mendatang
  - Digunakan untuk tujuan billing

# Sistem Operasi

---

- Bertanggungjawab mengelola sumber daya (*resources*)
- Berfungsi seperti software komputer biasa
  - Suatu program yang dieksekusi
- SO melepas kontrol & harus bergantung pada processor untuk mendapatkannya kembali

# SO sebagai *Resource Manager*

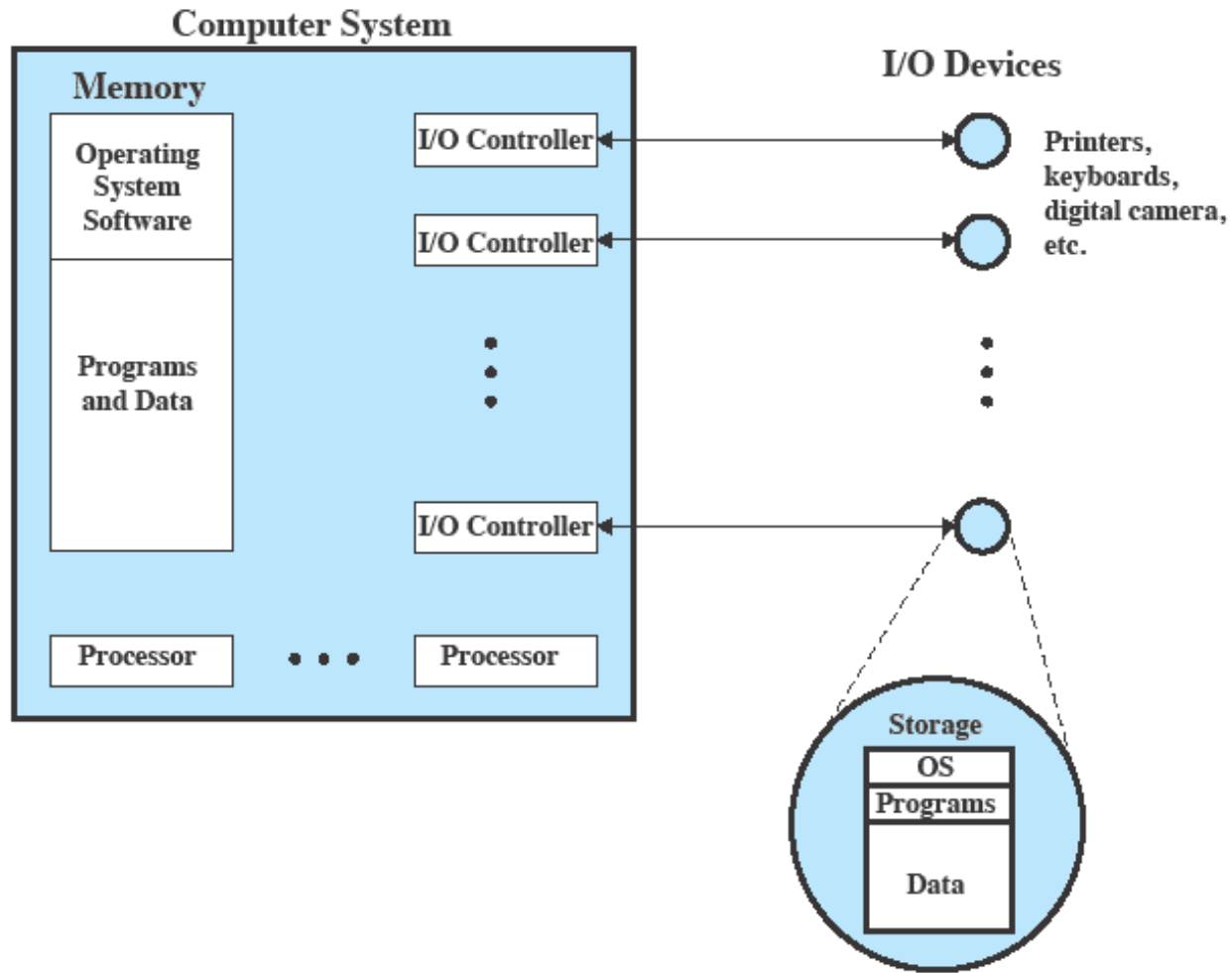


Figure 2.2 The Operating System as Resource Manager

# Kernel

---

- Bagian dari sistem operasi yang berada dalam memory utama
- Mengandung sebagian besar fungsi yang sering digunakan

# Evolusi Sistem Operasi

---

- Terjadi karena:
  - Upgrade hardware, ditambah tipe baru hardware
    - Device driver
  - Layanan baru
    - Sekarang: jarang
  - Perbaiki
    - Sebagian besar: perbaikan keamanan!

# **Evolusi Sistem Operasi**

... sejarah singkat ...

# Evolusi Sistem Operasi (1)

---

- Pemrosesan Serial
  - Bukan sistem operasi
  - Mesin berjalan dari suatu console dengan display light, toggle switches, input device, dan printer

# Evolusi Sistem Operasi (2)

---

- Pemrosesan Serial
  - Waktu jadwal
  - Setup mengikutkan pemuatan compiler, source program, menyimpan compiled program, dan memuat dan me-link-nya (linking)

# Evolusi Sistem Operasi (3)

---

- Sistem Batch Sederhana
  - Monitor
    - Software yang mengontrol urutan kejadian
    - Batch jobs bersama – bahasa kontrol job
    - Program mengembalikan kontrol ke monitor saat selesai
  - Dukungan hardware bagi model monitor
    - Proteksi memory: beberapa area memory dapat diakses hanya untuk monitor
    - Instruksi modus *privileged*: hanya dapat diakses ke monitor
    - Interupsi (mesin-mesin awal tidak punya ini)

# Contoh Utilisasi Sistem

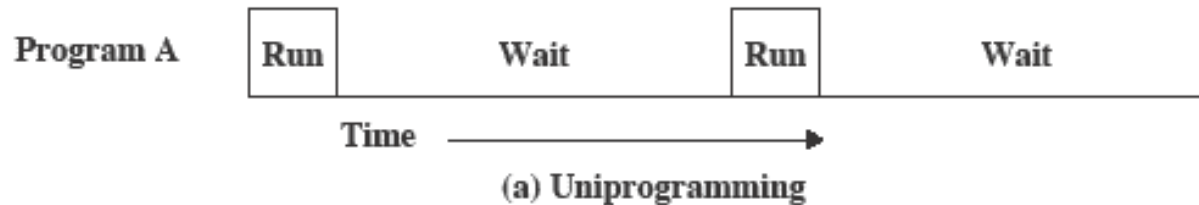
Read one record from file	15 $\mu$ s
Execute 100 instructions	1 $\mu$ s
Write one record to file	<u>15 <math>\mu</math>s</u>
TOTAL	31 $\mu$ s

Percent CPU Utilization =  $\frac{1}{31} = 0.032 = 3.2\%$

**Figure 2.4 System Utilization Example**

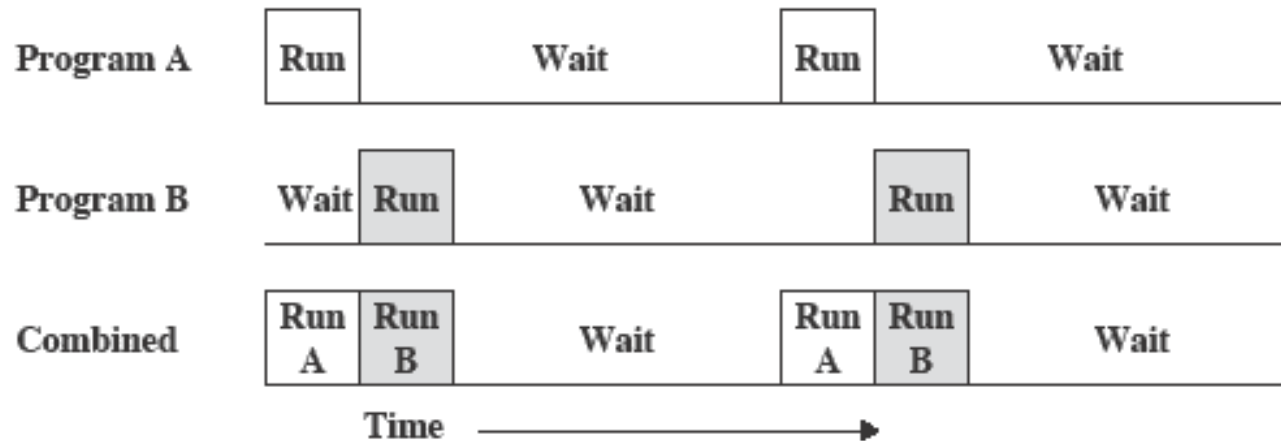
# Uniprogramming

- Processor harus menunggu instruksi I/O selesai sebelum bekerja lagi



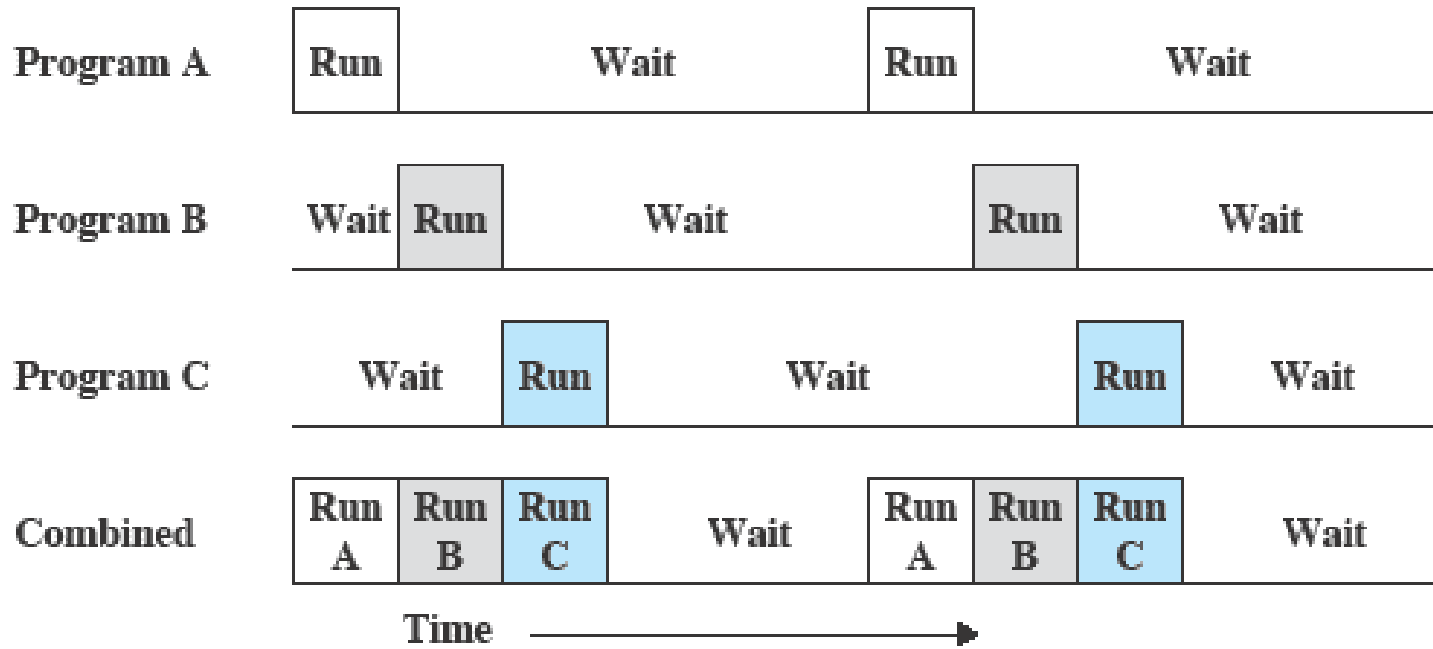
# Multiprogramming

- Ketika satu job perlu menunggu I/O, processor dapat berpindah ke job lain



(b) Multiprogramming with two programs

# Multiprogramming



(c) Multiprogramming with three programs

# Sistem Time Sharing

---

- Menggunakan *multiprogramming* untuk menangani banyak job interaktif
- Waktu processor dibagi (pakai) antar banyak pengguna
- Banyak pengguna secara simultan mengakses sistem melalui terminal

# Batch Multiprogramming vs. Time Sharing

**Table 2.3 Batch Multiprogramming versus Time Sharing**

	<b>Batch Multiprogramming</b>	<b>Time Sharing</b>
Principal objective	Maximize processor use	Minimize response time
Source of directives to operating system	Job control language commands provided with the job	Commands entered at the terminal

# Capaian Besar Sistem Operasi

# Kajian Teoritis Kritis

---

5 Titik Kritis (maju pesat):

- Proses
- Manajemen Memory
- Proteksi & Keamanan Informasi
- Penjadwalan & Manajemen sumber daya
- Struktur sistem

# Proses (1)

---

- Program yang sedang dieksekusi
- *Instance* dari program yang berjalan pada suatu komputer
- Entitas yang dapat didiseraahkan dan dieksekusi pada processor
- Suatu unit aktifitas yang dicirikan oleh
  - Serangkaian thread eksekusi
  - Suatu status terkini (current state)
  - Kumpulan berkait sumber daya sistem

# Proses (2)

---

## **Kesulitan Merancang Software Sistem** (dalam penanganan proses):

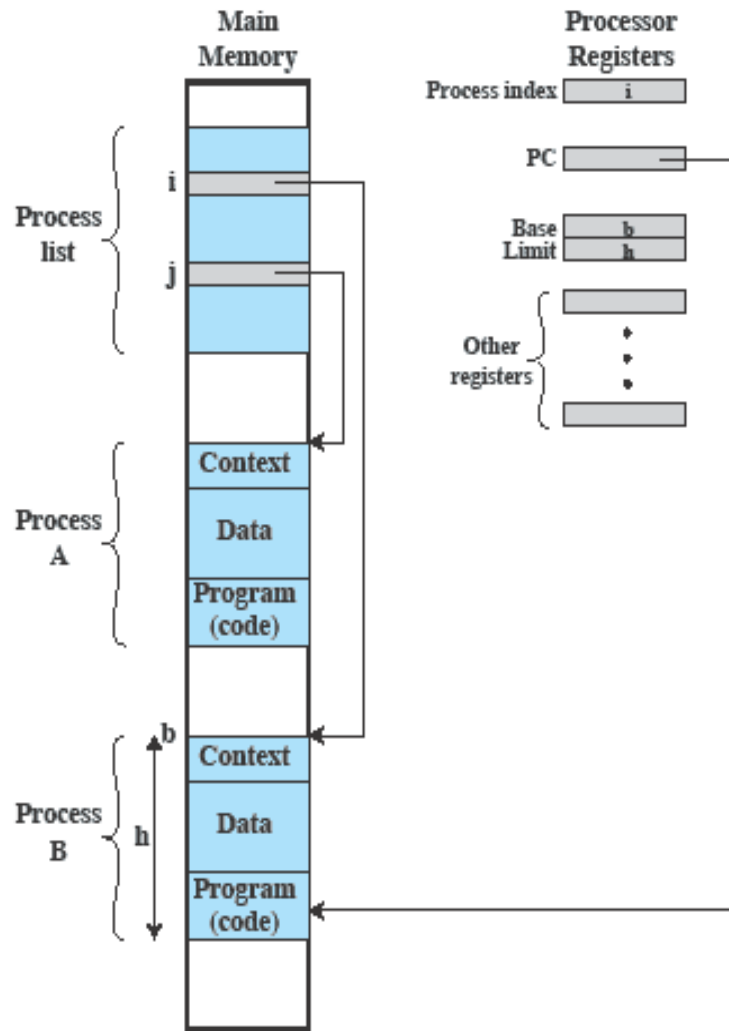
- Sinkronisasi tidak tepat
- Mutual exclusion (mutex) yang gagal
- Operasi program bersifat Nondeterminate
- Deadlocks

# Proses (3)

---

- Terdiri dari 3 komponen
  - Program yang *executable*
  - Data yang diperlukan oleh program
  - Konteks eksekusi dari program
    - Semua informasi yang diperlukan sistem operasi untuk mengelola proses tersebut

# Proses (4)



# Manajemen Memory

---

- Isolasi proses
- Alokasi dan manajemen otomatis
- Mendukung pemrograman modular
- Proteksi dan kendali akses
- Media simpan (storage) long-term

# Memory Virtual

---

- Mengizinkan *programmer* untuk mengamati memory dari suatu *point of view* logik
  - Layer lain dari indirection
- Membolehkan ilusi operasi dengan suatu ruang memory lebih besar daripada yang tersedia dalam kenyataan
  - Dengan menyimpan beberapa informasi pada sistem file

# Paging

---

- Satu cara untuk mewujudkan virtual memory
- Memungkinkan proses untuk disusun dari sejumlah blok *fixed-size*, dinamakan *page*
- Virtual address adalah suatu nomor *page* & *offset* di dalam page tersebut
- Setiap *page* dapat diletakkan dimana pun di dalam memory utama
- *Real address* atau *physical address* adalah alamat main memory

# Proteksi & Keamanan Informasi (1)

---

- Ketersediaan (*Availability*)
  - Fokus pada melindungi sistem terhadap interupsi (gangguan)
- Confidentiality
  - Memastikan bahwa pengguna tidak dapat membaca data yang tidak diotorisasi

# Proteksi & Keamanan Informasi (2)

---

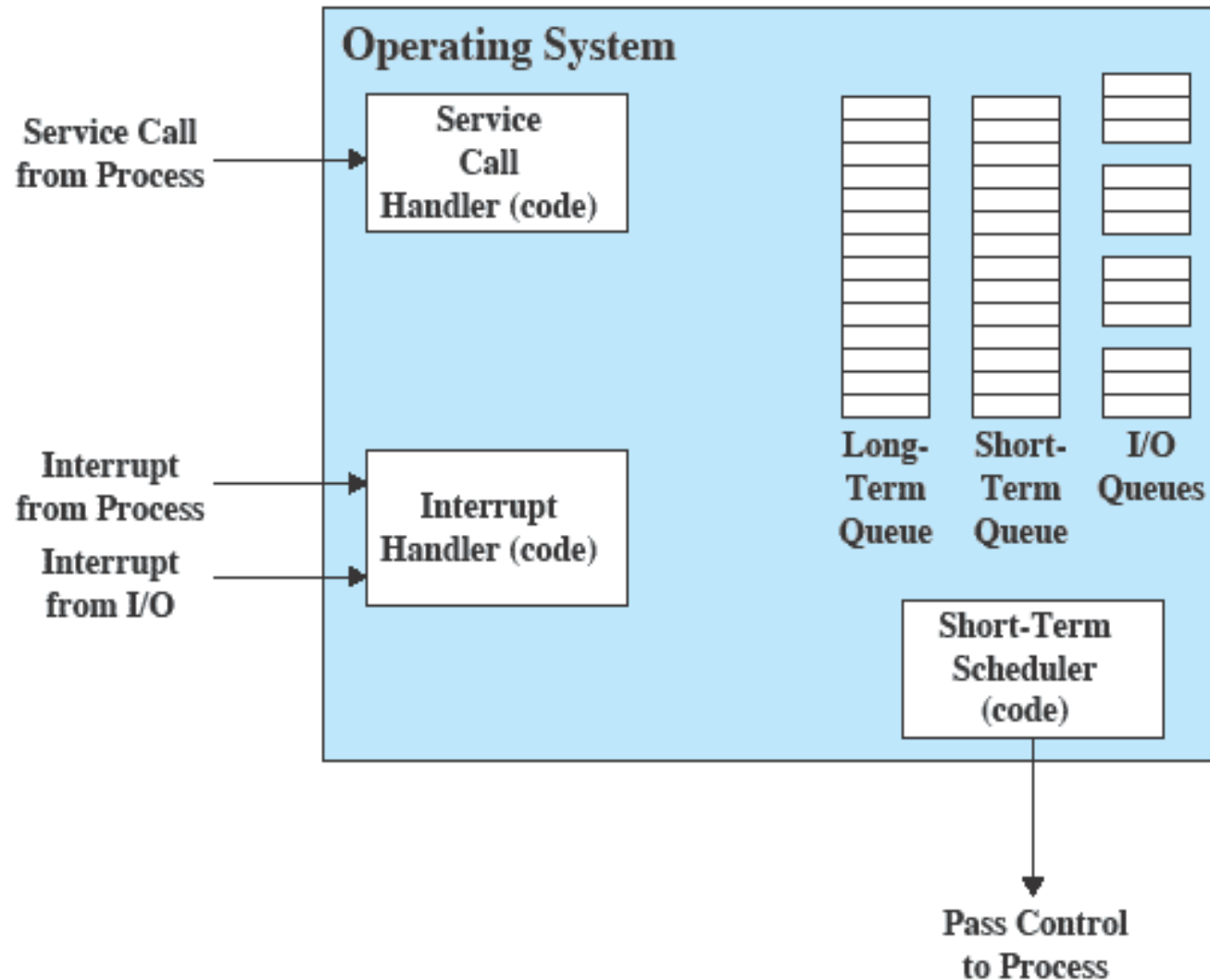
- Integritas Data
  - Melindungi data dari perubahan terlarang
- Otentikasi
  - Fokus pada verifikasi yang benar dari identitas pengguna & validitas pesan atau data

# Penjadwalan & Manajemen Resource

---

- Fairness
  - Memberikan akses yang adil dan sama terhadap sumber daya
- Differential responsiveness
  - Diskriminasi antara kelas job berbeda
- Efisiensi
  - Memaksimalkan throughput, meminimalkan response time, dan mengakomodasi sebanyak mungkin penggunaan

# Elemen Kunci dari SO



# Sistem Operasi Modern (1)

---

- Arsitektur Mikrokernel
  - Memberikan hanya sedikit fungsi esensi ke kernel
    - Address spaces
    - Interprocess communication (IPC)
    - Basic scheduling
  - Contoh: Mach, QNX
  - SO kini : Windows, Linux, Mac OS TIDAK berbasis mikrokernel

*“designing a monolithic kernel in 1991 is a fundamental error. Be thankful you are not my student. You would not get a high grade for such a design :-)”*

Andrew Tanenbaum to Linus Torvalds

# Sistem Operasi Modern (2)

---

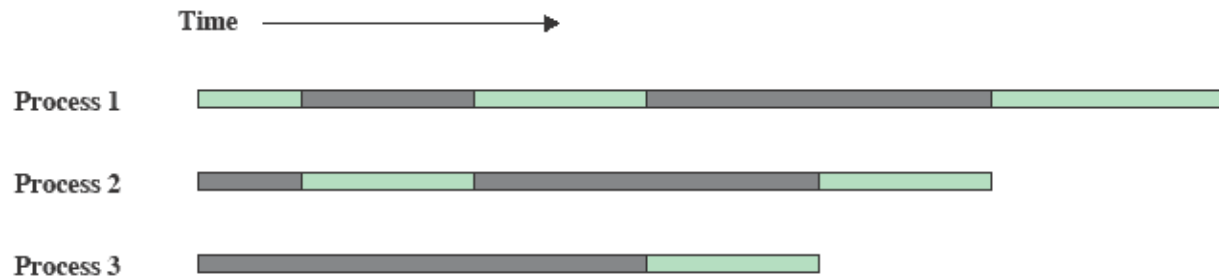
- Multithreading
  - Proses dibagi ke dalam thread-thread yang dapat berjalan secara konkuren
    - Thread
      - Unit kerja dispatchable
      - Eksekusi secara urut & interruptable
    - Proses merupakan koleksi satu atau lebih thread

# Sistem Operasi Modern (3)

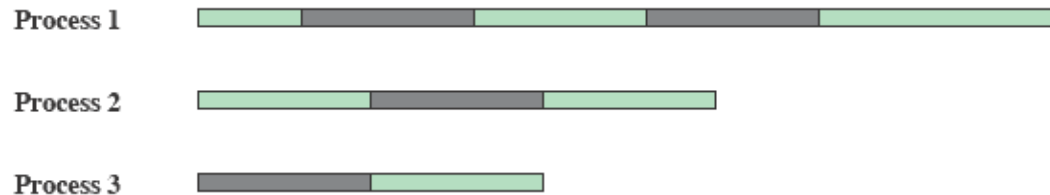
---

- Symmetric multiprocessing (SMP)
  - Terdapat beberapa processor
  - Processor-processor tersebut berbagi-pakai memory utama & fasilitas I/O yang sama
  - Semua processor dapat mengerjakan fungsi yang sama

# Multiprogramming & Multiprocessing



(a) Interleaving (multiprogramming, one processor)



(b) Interleaving and overlapping (multiprocessing; two processors)

Blocked Running

Figure 2.12 Multiprogramming and Multiprocessing

# Sistem Operasi Modern (4)

---

- Sistem Operasi Terdistribusi
  - Provides the illusion of a single main memory space and single secondary memory space
  - Eg. Amoeba by Andrew Tanenbaum

# Sistem Operasi Modern (5)

---

- Rancangan Berorientasi Obyek
  - Digunakan untuk penambahan ekstensi modular terhadap kernel kecil
  - Memungkinkan programmer meng-customize suatu SO tanpa mengganggu integritas sistem
- Contoh
  - NeXTSTEP – Objective C, di atas kernel BSD & Mach
  - BeOS – C++ membungkus suatu kernel C

# Tugas

---

- Baca sub bab 2.5 dan 2.8. Buat rangkuman mengenai kedua Sistem Operasi tersebut!
- Kerjakan soal Problems 2.1 dan 2.2!