**TUGAS PERTEMUAN 5**

**PENGANTAR SISTEM OPERASI**

***Resume Chapter 7 (Operating System by William Stallings)***



**Disusun oleh:**

**Rama Pramudya Wibisana 2022320019**

**PROGRAM STUDI SISTEM INFORMASI**

**FAKULTAS INFORMATIKA**

**UNIVERSITAS BINA INSANI**

**BEKASI**

**2023**

**CHAPTER 7**

**OPERATING SYSTEM (OS)**

1. **DEFINISI OS**

Operating System (OS) merupakan suatu program yang mengatur eksekusi program-program aplikasi dan berfungsi sebagai interface antara pengguna komputer dengan hardware komputer.

1. **TUJUAN DAN FUNGSI OS**
2. **Tujuan OS**
3. Kemudahan (OS Membuat komputer lebih mudah untuk digunakan)
4. Efisiensi (OS memungkinkan sumber daya sistem komputer digunakan dengan cara yang efisien)
5. Kemampuan berkembang (OS harus disusun sedemikian rupa sehingga memungkinkan pengembangan yang efektif)

**Layers dan Gambaran Sistem Komputer**

End User

Programmer

Application

Programs

Utilities

Operating System

Designer

Operating System

Computer hardware

1. **Fungsi OS**
2. Pembuatan program
3. Eksekusi Program
4. Akses ke perangkat I/O
5. Akses terkontrol ke file
6. Akses sistem
7. Deteksi error dan respons
8. Laporan
9. **TIPE OS**
10. Interaktif
11. Batch
12. Single program (uni-programming)
13. Multi-programming (Multi-tasking)
14. **Sistem-sistem Lama**
15. Akhir 1940 sampai pertengahan 1950
16. Tidak ada sistem operasi
17. Program berhubungan langsung dengan hardware
18. Terdapat dua masalah utama:

* Penjadwalan (Scheduling)
* Waktu setup (Setup time)

1. **Sistem Batch Sederhana**
2. Program resident monitor
3. Pengguna mengajukan job ke operator
4. Monitor mengontrol rangkaian event untuk memproses kumpulan job
5. Setiap job dibuat bercabang agar Kembali ke monitor apabila pengolahannya selesai, pada posisi ini monitor akan mulai memuatkan secara otomatis program berikutnya
6. Monitor menangani scheduling

* Monitor merupakan Teknik sistem operasi batch dengan menggunakan potongan software
* Job merupakan sebuah program tunggal
* Resident memory merupakan bagian monitor yang berada di memori utama

1. **Job Control Language (JCL)** 
   1. Instruksi untuk monitor
   2. Diawali oleh $
   3. e.g

* $JOB
* $FTN
* …. Some Fortran Instuctions
* $LOAD
* $RUN
* …. Some data
* $END

1. **Fitur Hardware Yang Diperlukan**
2. Proteksi memori

* Untuk melindungi monitor

1. Timer

* Untuk mencegah job memonopoli sistem

1. Instuksi istimewa (privileged instuctions)

* Hanya dieksekusi monitor
* e.g I/O

1. interrupts

* dibolehkan untuk melepaskan dan mendapatkan Kembali control

1. **Sistem Batch Multi-programmed** 
   1. Perangkat I/O sangat lambat
   2. Ketika sebuah program menunggu untuk I/O, program lain bisa menggunakan CPU
2. **Sistem Pembagian Waktu** 
   1. Mengizinkan Pengguna untuk berkomunikasi langsung dengan komputer

* i.e. Interactive
  1. Multi-programming mengizinkan sejumlah Pengguna untuk berkomunikasi dengan komputer

1. **Penjadwalan (Scheduling)** 
   1. Kunci untuk dapat multi-programming
   2. Penjadwalan Long term

* Ditetapkan dimana program diajukan untuk diproses
* Derajat pengontrolan multi-programming
* Saat diajukan, sebuah job akan diproses untuk penjadwalan short them
* Atau akan menukar job untuk penjadwalan medium term
  1. Penjadwalan Medium term
* Bagian dari fungsi penukaran
* Biasanya berdasarkan pada kebutuhan untuk mengatur multi-programming
* Jika tidak ada virtual memori, maka pengaturan memori juga sebuah isuue
  1. Penjadwalan Short term
* Pengatur pengiriman berita (Dispatcher)
* Fine grained decisions of which job to execute next
* i.e which job actually gets to use the processor in the next time slot
  1. I/O

1. **Gerbang Proses**

Release

Dispatch

Admit

Event Wait

Event Occurs

Timeout

**New**

**Ready**

**Blocked**

**Running**

**Exit**

1. **Process Control Block** 
   1. Identifikasi
   2. Gerbang
   3. Priori
   4. Program counter
   5. Memory pointers
   6. Data isi
   7. Status I/O
   8. Informasi laporan
2. **Managemen Memori** 
   1. Uni-program (program tunggal)

* Memori dipisah menjadi dua
* Satu untuk OS (monitor)
* Satu untuk eksekusi program
  1. Multi-program
* Pengguna merupakan bagian dari proses yang sedang aktif

1. **Swapping (Penukaran)** 
   1. Swapping merupakan antrian long term proses penyimpanan pada disk
   2. Swapping merupakan proses pada memori utama dalam keadaan ready (i.e semua I/O di blocked)

* OS akan memindahkan proses yang di block ke antrian, antrian ini adalah proses yang dikeluarkan sementara atau ditunda
* Pemindahan pada proses ready atau sebuah proses baru
* Tetapi swapping adalah proses I/O
  1. Masalah : I/O lebih lambat dibandingkanCPU dan CPU akan mempunyai

banyak waktu kosong (tidak bekerja)

* 1. Solusi :
* menambah memori utama
  + - Mahal
    - Diutamakan untuk program besar
* Swapping
* Memindahkan isi memori utama ke memori sekunder

1. **Pembagian (Partitioning)** 
   1. Membagi memori kedalam bagian-bagian untuk alokasi pemrosesan (termasuk OS)
   2. Fixed-sized partitions (partisi tetap)

* Membagi memori utama dengan ukuran yang tetap
* Setiap proses yang berukuran kecil atau sama dapat menempati sembarang partisi
* Jika partisi penuh, maka OS dapat men-swap beberapa proses dan memuatkan proses lain
* Tidak efisien dalam penggunaan memori karena adda fragmentasi internal, jumlah proses aktif tetap

1. **Partisi Dinamis** 
   1. Partisi yang digunakan memiliki Panjang dan jumlah yang berbeda – beda
   2. Proses dialokasikan ke dalam partisi yang ukurannya sama dengan proses
   3. Terdapat ruang kosong diakhir memori, terlalu kecil untuk digunakan

* Satu ruang kosong kecil – memori terbuang
  1. Ketika semua proses dibloked, OS akan men-swap sebuah proses dan menempatkan proses lain

1. **Variable Sized Partitions** 
   1. Proses baru ukurannya mungkin lebih kecil dari proses yang di-swap
   2. Akan terjadi ruang kosong lagi
   3. Akan menimbulkan banyak ruang kosong
   4. Solusi:

* Koalisi – menggabungkan semua ruang kosong menjadi ruang kosong yang besar
* Kompaksi – OS menggeser proses – proses menjadi berada dalam suatu blok, sehingga terdapat ruang kosong yang besar (c.f. disk de-fragmentation)
  1. Relokasi
* Tidak ada jaminan bahwa proses akan menempati partisi yang sama dalam memori
* Instruksi terdiri dari alamat
* Lokasi data
* Alamat data instuksi (branching)
* Alamat logika – referensi ke sebuah lokasi memori
* Alamat fisik – lokasi actual di dalam memori
  1. Paging
* Memori utama dibagi menjadi ukuran yang sama, chunk kecil -page frames
* Program dibagi menjadi proses – proses berukuran sama (chunk) – pages
* Mengalokasikan frame semua page ke sebuah proses
* OS menjaga daftar frame yang bebas
* Menggunakan page table untuk menunjukkan lokasi semua page proses

1. **Virtual Memory** 
   1. Tuntunan Paging

* Tidak memuat semua page proses di dalam memory
* Hanya memuat page seperti yang diminta
  1. Page Vault
* Page yang diinginkan tidak berada dalam memori
* OS harus men-swap page yang diinginkan
* Diperlukan untuk men-swap sebuah page agar terdapat ruang kosong
* Menyeleksi page yang dikeluarkan berdasarkan history

1. **Thrashing**

* Terlalu banyak proses pada memori yang kecil
* OS terlalu banyak melakukan swapping
* Kecil atau tidak ada proses yang selesai
* Disk menyala sepanjang waktu
* Solusi :
* Penggunaan algoritma penempatan yang baik
* Memperkecil jumlah proses yang bekerja

1. **Segmentasi**

* Paging adakalanya tidak berguna bagi programmer
* Segmentasi dimungkinkan bagi programmer
* Biasanya program dan data dialokasikakn di segment yang berbeda
* Segmentasi memungkinkan pemrograman menganggap memori terdiri dari kumpulan ruang alamat / segment
* Keunggulan segmentasi :
* Menyederhanakan penanganan perkembangan struktur data
* Memungkinkan program untuk diubah dan dikomplasi ulang secara independent, tanpa relinking dan memuat kembali seluruh program
* Membiarkan dirinya untuk berbagi pakai proteksi
* Memungkinkan dilakukan proteksi
* Beberapa sistem menggabungkan segmentasi dan paging