

Bab 1

Pengenalan Sistem Operasi

POKOK BAHASAN:

- ✓ Pengertian Sistem Operasi
- ✓ Perkembangan Sistem Operasi

TUJUAN BELAJAR:

Setelah mempelajari materi dalam bab ini, mahasiswa diharapkan mampu:

- ✓ Memahami definisi, maksud dan tujuan sistem operasi
- ✓ Memahami perkembangan sistem komputer dan sistem operasi

1.1 APAKAH SISTEM OPERASI ?

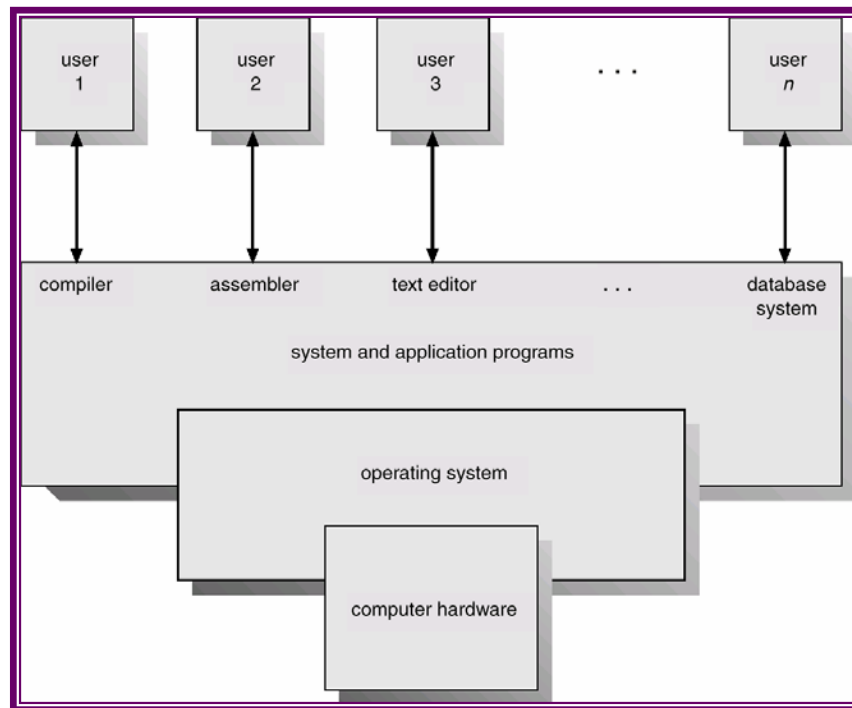
Sistem operasi adalah program yang bertindak sebagai perantara antara user dengan perangkat keras komputer. Sistem operasi digunakan untuk mengeksekusi program user dan memudahkan menyelesaikan permasalahan user. Selain itu dengan adanya sistem operasi membuat sistem komputer nyaman digunakan. Sistem operasi mempunyai tujuan untuk menggunakan perangkat keras komputer secara efisien.

Secara umum komponen sistem komputer terdiri dari :

1. **Perangkat Keras**, merupakan sumber daya utama untuk proses komputasi. Perangkat keras komputer terdiri dari : CPU, memory dan perangkat input output.

2. **Sistem Operasi**, mempunyai tugas untuk melakukan control dan koordinasi penggunaan perangkat keras pada berbagai program aplikasi untuk user-user yang berbeda.
3. **Program Aplikasi**, menentukan cara sumber daya sistem digunakan untuk menyelesaikan permasalahan komputasi dari user, contohnya compiler, sistem basis data, video games, program bisnis dan lain-lain.
4. **User** yang menggunakan sistem, terdiri dari orang, mesin atau komputer lain.

Hubungan antara komponen-komponen sistem komputer diatas dapat dilihat pada Gambar 1-1.



Gambar 1-1: Komponen-komponen sistem komputer

Sistem operasi didefinisikan sebagai :

- **Resource allocator**
Sistem operasi mengatur dan mengalokasikan sumber daya – sumber daya sistem komputer
- **Program control**

Sistem operasi melakukan control eksekusi dari program user dan operasi input output.

- **Kernel**

Sistem operasi sering disebut kernel, yaitu suatu program yang berjalan sepanjang waktu (selain program aplikasi).

1.2 SISTEM MAINFRAME

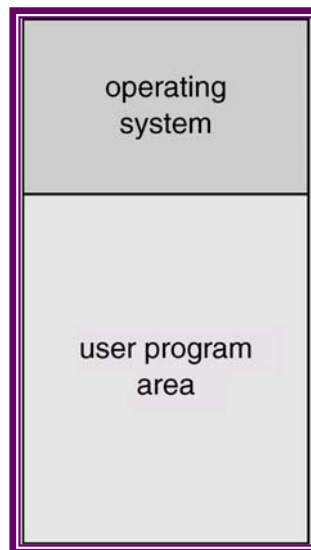
Sistem komputer pendahulu secara fisik berbentuk mesin besar yang disebut sistem mainframe. Untuk menjalankan sistem ini dilakukan dari suatu *console*. Perangkat input yang digunakan berupa *card reader* dan *tape drive*. Perangkat output yang digunakan berupa *line printer*, *tape drive* dan *card punch*. Kemudian, user menyiapkan job yang terdiri dari program, data dan beberapa informasi kontrol (*control card*) dan dikirimkan ke operator komputer. Job biasanya dalam bentuk *punch card*. Beberapa waktu kemudian (dalam hitungan waktu menit, jam atau hari), output ditampilkan. Output berupa hasil program, apabila terjadi error pada program memory dan register akan berisi kosong.

Sistem operasi pada komputer mainframe sangat sederhana. *Task* utama mengirim control secara otomatis dari satu *job* ke *job* berikutnya. Sistem operasi selalu residen di memory yang disebut dengan *resident monitor*. Gambar 1-2 adalah gambaran layout memori pada sistem *batch* sederhana.

Untuk meningkatkan kecepatan proses, job yang sama perlu dikumpulkan bersama (*batch*) dan dijalankan oleh komputer sebagai satu kelompok. Kemudian programmer memberikan program kepada operator. Operator akan mengurut program yang sama dan kemudian komputer akan menjalankan setiap kumpulan program tersebut. Output dari setiap job dikirim kembali kepada programmer.

Untuk menghindari adanya waktu nganggur CPU yang cukup lama maka dikembangkan suatu teknik mengurutan kerja *job* secara otomatis. Teknik ini mampu mentrasfer control secara otomatis dari suatu *job* ke *job* berikutnya. Inilah bentuk sistem operasi pertama kali. Program kecil yang bersifat residen di memori berisi urutan-urutan *job* yang akan berpindah secara otomatis inilah yang disebut dengan *Resident Monitor*.

Jika komputer dinyalakan, maka sistem akan menunjuk ke *resident monitor*, secara otomatis kontrol akan menunjuk ke program tersebut.

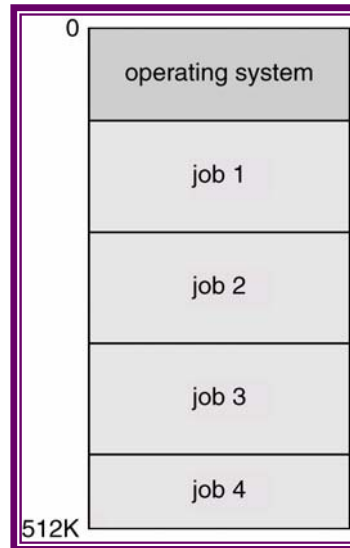


Gambar 1-2: Memory layout pada simple batch system

1.3 SISTEM BATCH MULTIPROGRAM

Beberapa *job* dikumpulkan oleh sistem operasi pada memory utama pada waktu yang sama, seperti pada Gambar 1-3. Kumpulan job ini merupakan bagian dari job yang disimpan pada pool (*job pool*). *Job pool* berisi job-job yang sudah siap dieksekusi. Jumlah job dapat disimpan bersama-sama pada memory biasanya lebih kecil daripada jumlah job yang dapat berada pada *job pool*. Sistem operasi mengambil beberapa *job* yang siap untuk dieksekusi untuk diletakkan di memori utama. Jika *job* yang sedang dieksekusi menunggu beberapa *task* (seperti proses *mount* tape drive atau operasi I/O yang harus diselesaikan), maka *job* tersebut diganti dengan *job* berikutnya.

Pada sistem multiprogramming, sistem operasi harus menyediakan mekanisme untuk manajemen memori, penjadwalan CPU dan manajemen disk. Sistem operasi multiprogram menyediakan supply untuk *I/O routine*. Sistem harus dapat mengalokasikan memory untuk beberapa job. Beberapa job yang sudah siap dieksekusi akan dipilih oleh sistem job mana yang akan dieksekusi oleh CPU. Perangkat apa saja yang diperlukan oleh setiap job juga harus dialokasikan oleh sistem.



Gambar 1-3: Memory layout pada sistem batch multiprogram

1.4 SISTEM TIME SHARING

Time sharing atau multitasking adalah pengembangan dari sistem multiprogram. Beberapa job yang berada pada memory utama dieksekusi oleh CPU secara bergantian. CPU hanya bisa menjalankan program yang berada pada memory utama. Perpindahan antar job terjadi sangat sering sehingga user dapat berinteraksi dengan setiap program pada saat dijalankan. Suatu job akan dipindahkan dari memori ke disk dan sebaliknya.

Sistem *time sharing* juga disebut dengan sistem komputasi interaktif, dimana sistem komputer menyediakan komunikasi on-line antara user dengan sistem. User memberikan instruksi pada sistem operasi atau program secara langsung dan menerima respon segera. Perangkat input berupa keyboard dan perangkat output berupa *display screen*, seperti *cathode-ray tube* (CRT) atau monitor. Bila sistem operasi selesai mengeksekusi satu perintah, maka sistem akan mencari pernyataan berikutnya dari user melalui keyboard. Sistem menyediakan editor interaktif untuk menulis program dan sistem debug untuk membantu melakukan debugging program.

Agar user dapat mengakses data dan kode program dengan nyaman, sistem menyediakan sistem file online. Suatu file adalah kumpulan informasi yang berhubungan yang didefinisikan oleh pembuatnya. Biasanya, file berupa program (baik

bentuk *source* dan *object*) dan data. Data file berupa teks dengan format tertentu. Secara umum, file adalah kumpulan bit, bite, baris atau *record*. Sistem operasi mengimplementasikan konsep abstrak dari file dengan mengatur perangkat penyimpanan seperti tape dan disk. File secara normal diorganisasikan dalam logical cluster atau directory, untuk memudahkan lokasi dan akses file.

1.5 SISTEM DESKTOP

Semakin turunnya harga perangkat keras, dikembangkan sistem komputer untuk satu user. Jenis sistem komputer ini biasanya disebut dengan *personal computer* (PC). Perangkat I/O berupa keyboard dan mouse, dan perangkat output berupa display screen atau printer yang berkecepatan tinggi.

Personal komputer dikembangkan tahun 1970-an. Sistem ini disebut dengan mikrokomputer. Sistem operasi masih belum dikembangkan untuk multiuser maupun multitasking. Tujuan sistem operasi adalah untuk memaksimalkan utilitas CPU dan *peripheral*, serta memaksimalkan kenyamanan dan respon user. Sistem operasi yang dikembangkan adalah Microsoft Windows dan Apple Machintosh. Sistem operasi MS-DOS dari Microsoft yang masih single tasking dikembangkan oleh IBM menjadi OS/2 yang merupakan sistem multitasking.

Berkembangnya sistem komputer dari mainframe menjadi mikrokomputer menunjukkan bahwa sistem operasi mikrokomputer dapat mengadopsi sistem mainframe.

Contoh perpindahan sistem operasi adalah perkembangan sistem operasi MULTICS. MULTICS dikembangkan tahun 1965 sampai 1970 oleh Massachusetts Institute of Technology (MIT) sebagai utilitas komputasi yang berjalan pada komputer mainframe yang besar dan kompleks. Kemudian Beel Laboratories mengembangkan MULTICS dengan mendesain UNIX tahun 1970 untuk minikomputer PDP-11. Tahun 1980, dikembangkan sistem operasi UNIX-like untuk sistem mikrokomputer menyusul sistem operasi lain yaitu Microsoft Windows NT, IBM OS/2 dan Machintosh.

1.6 SISTEM PARALEL

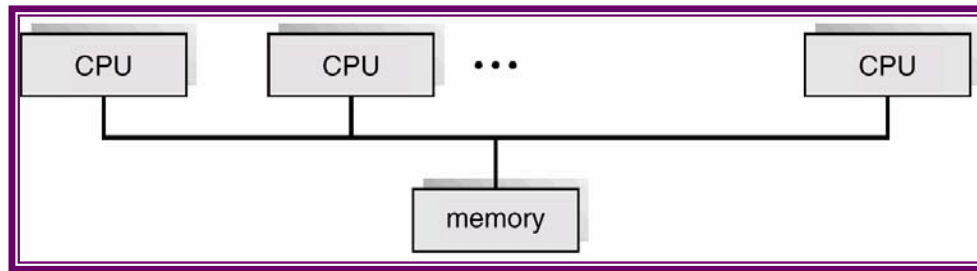
Sistem paralel atau sistem multiprocessor mempunyai lebih dari satu prosesor yang dapat berkomunikasi, membagi *bus*, *clock* dan juga perangkat memory dan peripheral. Sistem ini disebut sebagai *tightly coupled system*.

Sistem ini dikembangkan karena beberapa alasan. Salah satu keuntungan dari sistem ini adalah meningkatkan jumlah proses yang dapat dijalankan pada satu waktu (*throughput*). Dengan meningkatkan jumlah prosesor, diharapkan pekerjaan dapat dikerjakan dalam waktu yang lebih pendek.

Alasan lain dari pengembangan sistem multiprocessor adalah meningkatkan kehandalan sistem. Jika fungsi dapat didistribusikan pada beberapa prosesor, maka kegagalan dari satu prosesor tidak akan menghentikan sistem, tetapi hanya memperlambat sistem. Jika terdapat 10 prosesor dan satu gagal, maka sisa 9 prosesor menggantikan pekerjaan prosesor yang gagal. Keseluruhan sistem hanya memperlambat 10 persen. Kemampuan untuk melanjutkan penyediaan layanan untuk menyelamatkan perangkat keras disebut *gracefull degradation*. Sistem yang didesain untuk *gracefull degradation* juga disebut *faul- tolerant*.

Sistem multi prosesor yang sering digunakan adalah model *symmetric multiprocessing*, dimana setiap prosesor menjalankan sistem operasi yang identik dan komunikasi antar prosesor jika diperlukan. Beberapa sistem menggunakan *asymmetric multiprocessing*, dimana setiap prosesor mempunyai tugas tertentu. Prosesor master mengontrol sistem, prosesor lain menunggu instruksi master atau mempunyai tugas yang ditentukan oleh master. Skema ini merupakan hubungan master-slave. Prosesor master menjadwalkan dan mengalokasikan pekerjaan dari prosesor slave.

Contoh *symmetric multiprocessing* adalah sistem UNIX versi Encore's untuk komputer Multimax. Komputer dapat dikonfigurasi untuk menangani satu lusin prosesor, semua menjalankan UNIX. Keuntungan dari model ini adalah bahwa beberapa proses dapat berjalan pada satu waktu (N proses jika terdapat N CPU) tanpa menyebabkan pengurangan performansi. Sehingga kita dapat mengontrol I/O secara hati-hati untuk menjamin data mendapatkan prosesor yang tepat. Arsitektur dari *symmetric multiprocessing* dapat dilihat pada Gambar 1-4.



Gambar 1-4: Arsitektur symmetric multiprocessing

1.7 SISTEM TERDISTRIBUSI

Tren sistem komputer saat ini adalah mendistribusikan komputasi diantara beberapa prosesor. Prosesor berkomunikasi dengan prosesor lain melalui saluran komunikasi, misalnya bus kecepatan tinggi atau saluran telepon. Sistem ini disebut *loosely coupled system* atau sistem terdistribusi (*distributed system*).

Prosesor pada sistem terdistribusi bervariasi ukuran dan fungsinya. Biasanya terdiri dari mikroprosesor, workstation, minikomputer dan sistem komputer *general-purpose*. Prosesor-prosesor ini disebut dengan site, node, komputer atau lainnya.

Keuntungan dari sistem terdistribusi adalah :

- *Resource sharing*

Jika sejumlah site yang berbeda dihubungkan, maka user pada site satu dapat menggunakan sumber daya dari site lainnya. Sebagai contoh, user pada site A dapat menggunakan printer laser dari site B. Sebaliknya user B dapat mengakses file user A.

- **Meningkatkan kecepatan komputasi**

Jika komputasi tertentu dapat dipartisi dalam sejumlah sub komputasi yang dapat berjalan secara konkuren, maka sistem terdistribusi dapat mendistribusikan komputasi pada beberapa site untuk menjalankan komputasi secara konkuren.

- **Lebih handal**

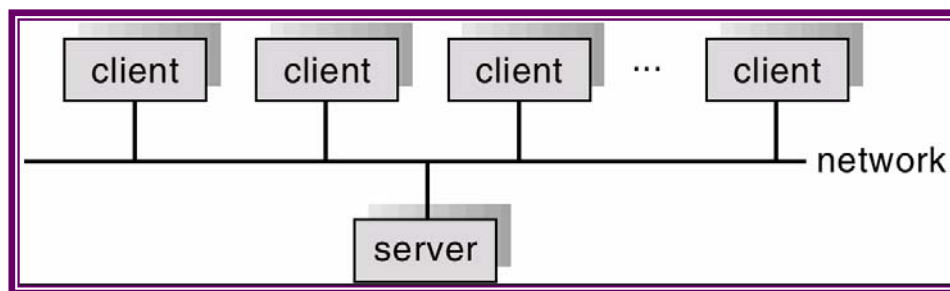
Jika satu site gagal pada sistem terdistribusi, sisa site dapat melanjutkan operasinya. Jika sistem dibagi sejumlah instalasi besar, maka kegagalan salah satunya tidak berakibat pada sisa sistem. Sebaliknya, jika sistem dibagi dalam sejumlah mesin kecil, masing-masing bertanggung jawab pada fungsi sistem yang penting (misalnya

terminal karakter I/O atau sistem file), maka satu kegagalan dapat menghentikan operasi dari keseluruhan sistem. Secara umum, jika terjadi redundansi pada sistem (baik perangkat keras maupun perangkat lunak), sistem dapat menjalankan operasinya meskipun beberapa site gagal.

- **Komunikasi**

Terdapat beberapa anggota program yang memerlukan mengganti data dengan data lain pada satu sistem. Sistem Windows contohnya, sering terjadi membagi data atau transfer data antara display. Jika beberapa site dihubungkan dengan lainnya dengan jaringan komunikasi, processor pada site yang berbeda dapat menukar informasi. User melakukan transfer file atau komunikasi dengan user lain melalui *electronic mail*. Seorang user dapat mengirim mail ke user lain pada site yang sama atau site yang berbeda.

Sistem terdistribusi memerlukan infrastruktur jaringan, berupa *local area network* (LAN) atau *wide area network* (WAN). Sistem terdistribusi biasanya disebut dengan sistem client-server atau peer-to-peer. Arsitektur dari sistem client server dapat dilihat pada Gambar 1-5.



Gambar 1-5: Arsitektur sistem client server

1.8 SISTEM TERKLASTER

Sistem terklaster (*clustered system*) adalah pengembangan dari sistem terdistribusi. Perbedaan sistem terklaster dengan sistem terdistribusi adalah pada sistem terklaster memungkinkan dua atau lebih sistem untuk membagi penyimpanan sekunder (*storage*) bersama-sama. Sistem ini mempunyai kehandalan sistem yang tinggi seperti pada sistem terdistribusi.

Sistem terkluster dapat berupa model *asymmetric clustering* dimana satu server menjalankan aplikasi sementara server lainnya standby. Model lainnya adalah *symmetric clustering* dimana semua host menjalankan aplikasi.

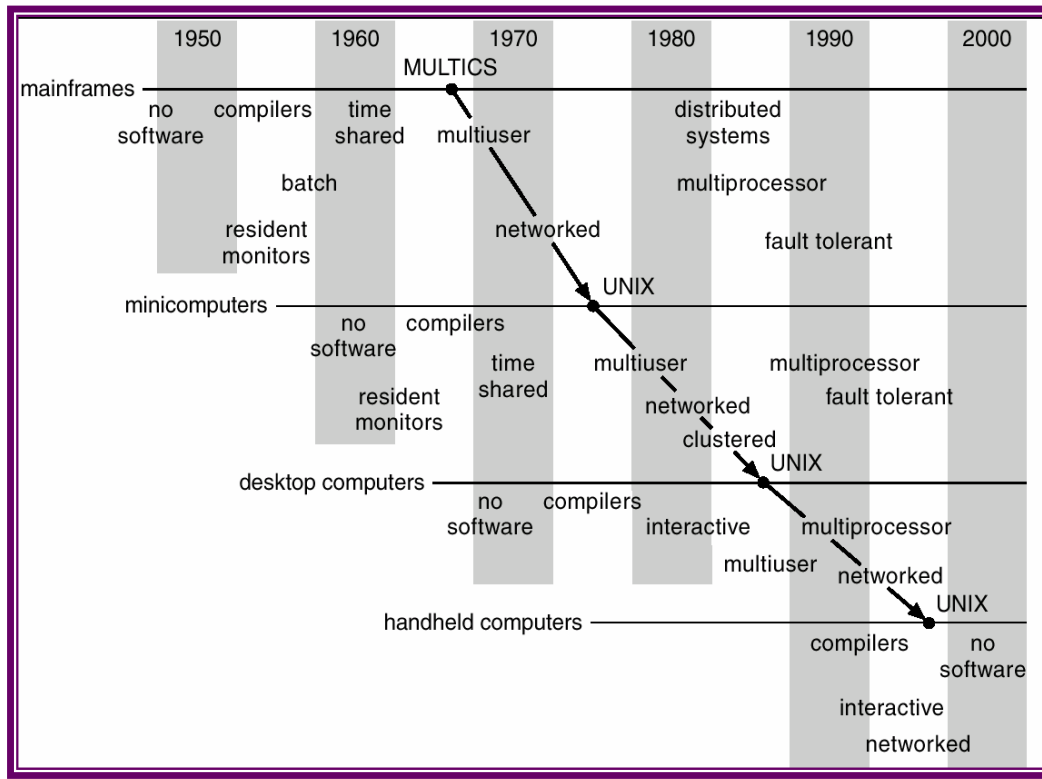
1.9 SISTEM REAL TIME

Salah satu bentuk sistem operasi untuk keperluan khusus adalah sistem *real time*. Sistem *real time* digunakan bila terdapat kebutuhan ketepatan waktu pada operasi prosesor atau aliran data sehingga sering digunakan untuk perangkat control pada suatu aplikasi seperti mengontrol percobaan keilmuan, sistem medical imaging, sistem control industri dan beberapa sistem display. Pada sistem *real time* harus didefinisikan batasan waktu yang tetap. Pemrosesan harus dikerjakan dalam waktu tertentu atau sistem akan gagal. Sebagai contoh, jika lengan robot tidak diinstruksikan untuk berhenti segera maka dapat merusak robot tersebut.

Terdapat dua bentuk sistem *real time*. Sistem *hard real time* menjamin tugas kritis diselesaikan tepat waktu. Pada sistem ini penyimpanan sekunder terbatas atau tidak digunakan, data langsung dikirim ke memory atau *read-only memory* (ROM) dalam waktu singkat. Pada sistem *hard real time* terjadi konflik pada sistem time sharing dan tidak didukung oleh sistem operasi tujuan umum. Bentuk lainnya adalah *soft real time* dimana tugas kritis mendapatkan prioritas lebih tinggi dari tugas lain dan setelah satu task selesai maka task berprioritas ini akan diselesaikan. Sistem ini terbatas pada industri pengontrol robot. Sangat berguna pada aplikasi multimedia dan virtual reality yang membutuhkan fitur sistem operasi tertentu.

1.10 SISTEM HANDHELD

Sekitar tahun 1990-an dikembangkan sistem yang lebih kecil dari mikrokomputer yang disebut dengan sistem *handheld* dalam bentuk *personal digital assistants* (PDA). Pada beberapa sistem terdapat telepon selular. Sistem ini mempunyai memory yang terbatas, prosesor dengan kecepatan rendah dan display screen yang kecil. Perkembangan sistem komputer dari sistem mainframe sampai *handheld* dan perkembangan sistem operasi dapat dilihat pada Gambar 1-6.



Gambar 1-6: Migrasi sistem komputer dan sistem operasi

LATIHAN SOAL :

1. Apa yang dimaksud sistem :
 - a. Batch system
 - b. Multiprogramming
 - c. Time sharing
 - d. Paralel (multiprocessor)
 - e. Terdistribusi
 - f. Cluster
 - g. *Real time*

h. *Handheld*

2. Apa perbedaan sistem *single programming* dan multiprogramming ?
3. Apa perbedaan *symmetric* dan *asymmetric multiprocessing* ?
4. Apa perbedaan sistem paralel dan sistem terdistribusi
5. Perbedaan sistem terdistribusi dan sistem terkluster ?
6. Apa yang dimaksud *symmetric* dan *asymmetric clustering* ?
7. Sebutkan keuntungan menggunakan sistem terdistribusi