



# **MODUL SISTEM OPERASI**

## **PENGELOLAAN MEMORI TANPA PAGING**

# FUNGSI MEMORI

- Mengelola informasi memori yang dipakai dan tidak sedang dipakai
- Mengalokasikan memori ke proses yang diperlukan
- Menddealokasikan memori dari proses yang telah selesai
- Mengelola swapping antara memori utama dan disk

# MANAJEMEN MEMORI PADA SISTEM MULTI-PROGRAMMING

Sistem operasi harus mendukung dua kebutuhan yang saling bertentangan yaitu

- Pemisahan ruang-ruang alamat
- Pemakaian bersama memori

# MANAJEMEN MEMORI BERDASARKAN KEBERADAAN SWAPPING

- Manajemen tanpa swapping (Manajemen memori tanpa pemindahan citra proses antara memori utama dan disk selama eksekusi)
- Manajemen dengan swapping (Manajemen memori dengan pemindahan citra proses antara memori utama dan disk selama eksekusi)

# Strategi Alokasi Memori

## 1. First-fit Algorithm

Manajer memori menelusuri peta bit atau senarai berkait sampai menemukan lubang besar yang memadai untuk ditempati proses. Lubang dibagi dua, untuk proses dan lubang yang tidak digunakan kecuali ketika besar lubang tepat sama dengan ukuran yang diperlukan proses.

## 2. Next-fit Algorithm

Penelusuran sama dengan first-fit algorithm, namun dimulai dari posisi terakhir kali menemukan segmen untuk proses.

# Strategi Alokasi Memori

## 3. Best-fit Algorithm

Algoritma mencari sampai akhir dan mengambil lubang terkecil yang dapat memuat proses.

## 4. *Worst-fit algorithm*

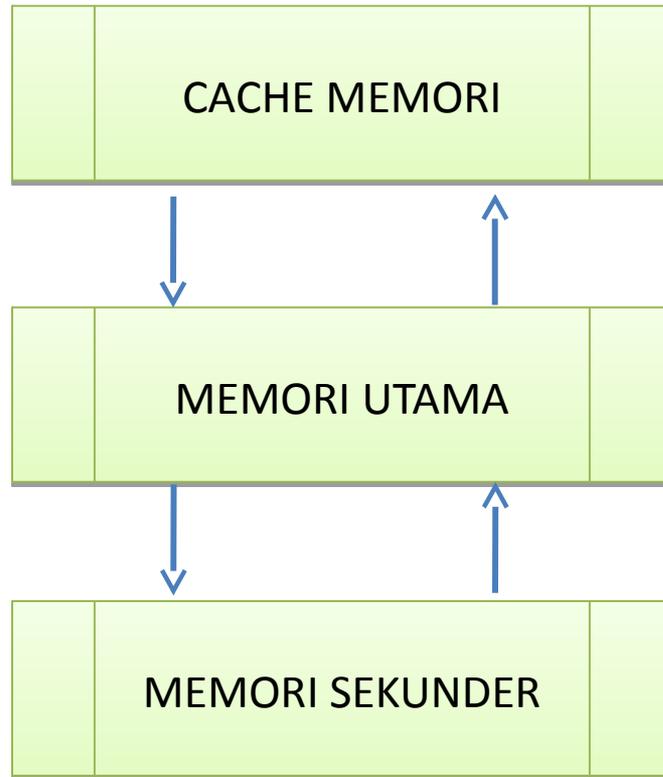
Selalu mencari lubang besar yang tersedia.

## 5. *Quick-fit algorithm*

Hanya untuk pencatatan pada senarai berkait.

Hanya menelusuri senarai lubang memori bukan di senarai proses.

# HIRARKI MEMORI



# HIRARKI MEMORI

- Pemakaian memori 2 tingkat menggunakan cache memori dapat meningkatkan kinerja dan utilisasi memori secara dramatik. Cache memori lebih mahal dibanding memori utama sehingga chace relatif kecil

# MANAJEMEN MEMORI TANPA SWAPPING

- Monoprograming
- Multiprogramming dengan pemartisian statis

# MANAJEMEN MEMORI MONO PROGRAMMING

- Hanya 1 proses pada satu saat
- Hanya satu proses menggunakan semua materi
- Pemakai memuatkan program ke seluruh memori dari disk atau tape
- Program mengambil kendali semua mesin

# MULTI-PROGRAMMING DENGAN PEMARTISIAN STATIS

- Mempermudah pemogram
- Agar dapat memberikan layanan interaktif ke beberapa orang secara simultan
- Efisiensi penggunaan sumber daya
- Eksekusi lebih mudah jika proses besar dipecah menjadi beberapa proses kecil
- Dapat mengerjakan sejumlah job secara simultan

# PEMARTISIAN MENJADI PARTISI-PARTISI BERUKURAN SAMA

- Beberapa proses yang ukurannya kurang atau sama dengan ukuran partisi dimasukkan ke sembarang partisi yang berbeda
- Kekurangannya bila program ukurannya lebih besar dari dibanding partisi yang tersedia maka program tidak dapat dimuatkan.
- Kekurangannya juga untuk program yg ukurannya kecil dibanding ukuran partisi maka banyak ruang yang tidak terpakai.

# STRATEGI PENEMPATAN PADA PARTISI-PARTISI YANG UKURANNYA BERBEDA

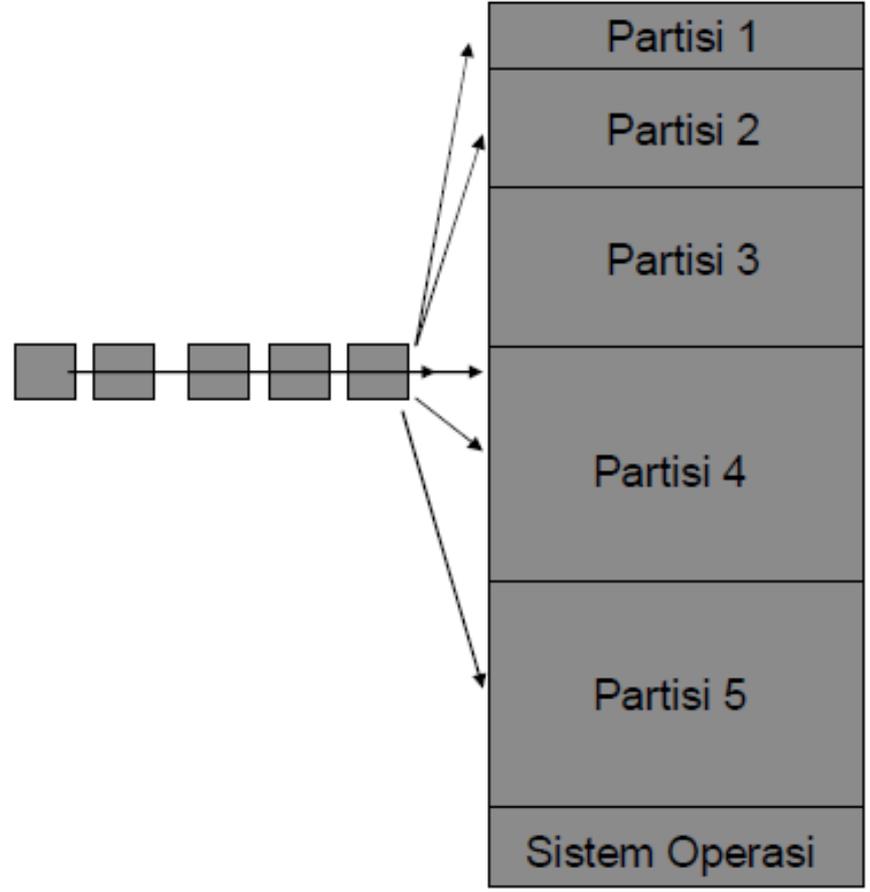
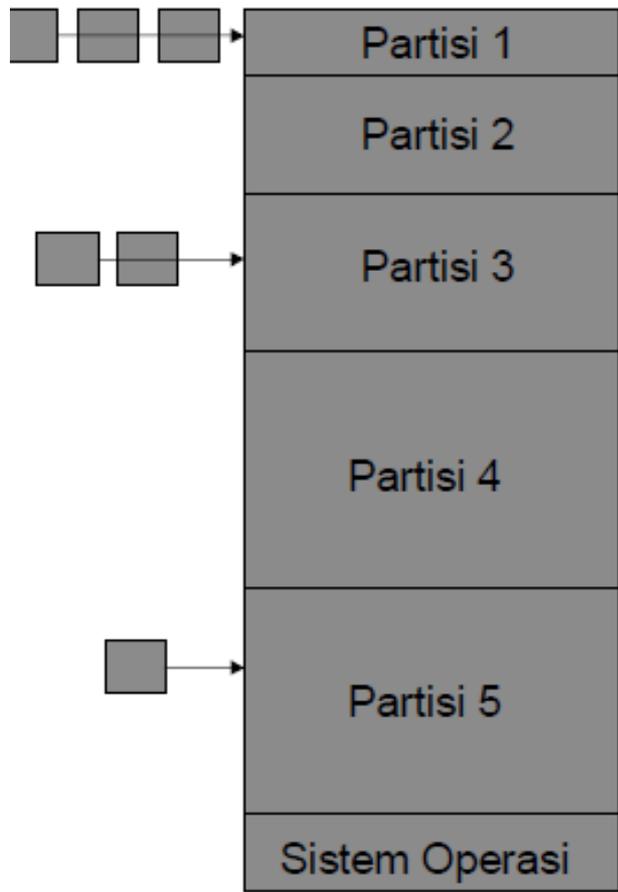
- Satu antrian untuk tiap partisi
- Satu antrian untuk semua partisi

# SATU ANTRIAN UNTUK TIAP PARTISI

- Proses ditempatkan ke partisi yang paling kecil yang dapat memuatnya
- Keuntungan: meminimalkan pemborosan memori
- Kelemahan: dapat terjadi antrian panjang disuatu partisi sementara antrian partisi2 yang lain kosong

# SATU ANTRIAN UNTUK TIAP PROSES

- Proses-proses diantrikan tunggal untuk semua partisi. Proses segera ditempatkan di partisi bebas paling kecil yang dapat memuat
- Keunggulan : lebih fleksibel serta implementasi dan operasi lebih minimal karena hanya mengelola satu antrian
- Kekurangan : proses dapat ditempatkan dipartisi yang banyak diboroskan, yaitu proses kecil ditempatkan di partisi yang sangat besar



## **KUIS :**

- 1. Jelaskan cara kerja sistem Mono Programming dan Multi Programming?**
- 2. Jelaskan tentang apa yang dimaksud dengan pengelolaan memori tanpa paging?**
- 3. Jelaskan secara lengkap strategi alokasi memori Best-fit Algorithm?**