

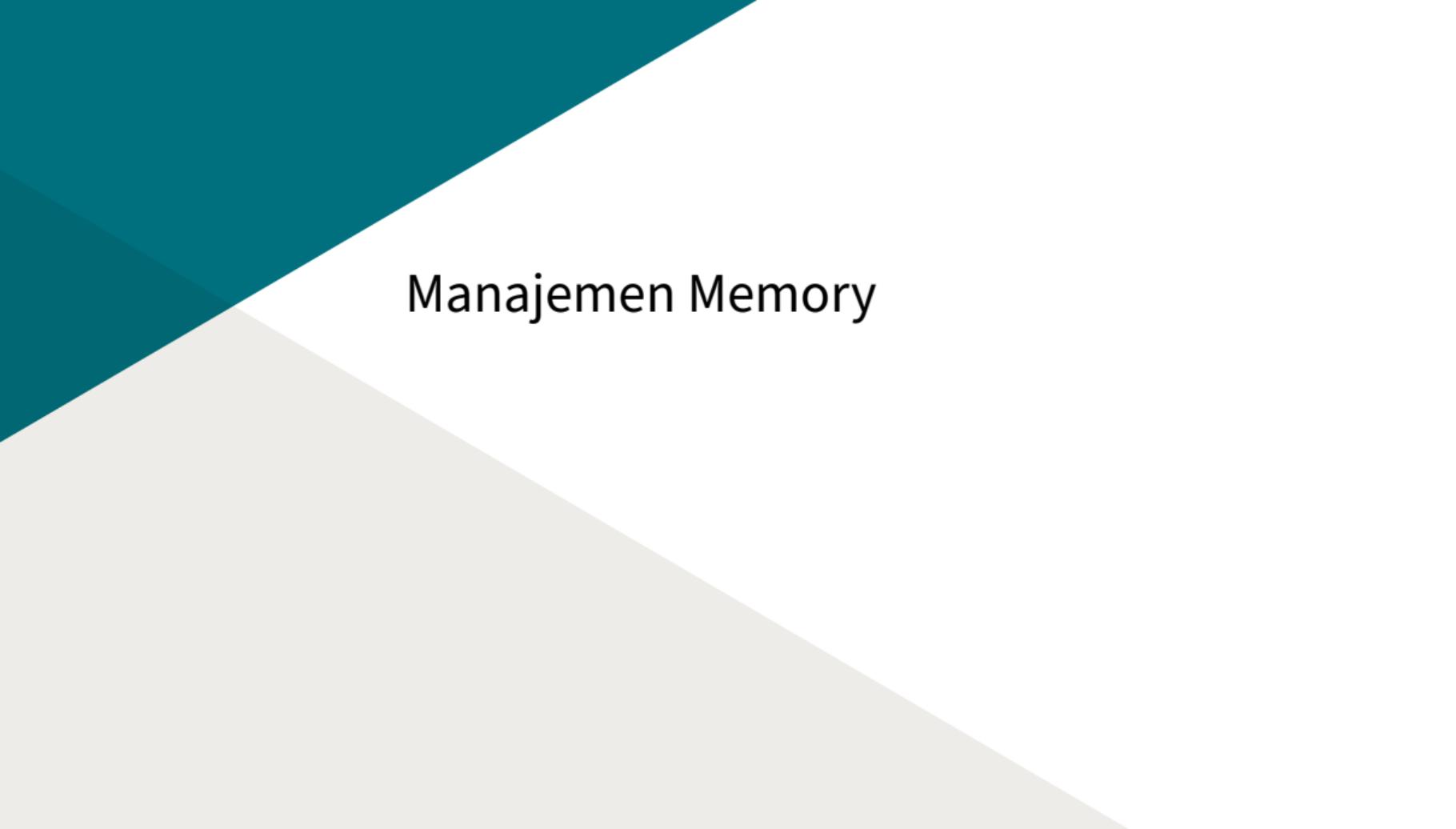


# Sistem Operasi

## Catatan Kuliah #10

Alauddin Maulana Hirzan, M. Kom

0607069401

The background consists of two large, overlapping geometric shapes. A teal-colored shape is in the upper-left corner, and a light gray shape is in the lower-left corner. The rest of the background is white. The text 'Manajemen Memory' is centered in the white area.

# Manajemen Memory



# Manajemen Memory

## Apa itu **Memory**?

Memori adalah sumber daya yang sangat penting dalam sistem komputer mana pun, dan manajemen memori yang efisien sangat penting untuk kelancaran sistem operasi. Memori mengacu pada komponen perangkat keras fisik yang menyimpan data dan instruksi yang dapat diakses dengan cepat oleh CPU (Central Processing Unit).

Data dapat diakses jauh lebih cepat daripada data yang disimpan pada disk drive atau perangkat penyimpanan sekunder lainnya. Instruksi dibaca ke dalam CPU oleh unit pengambilan instruksi, diterjemahkan oleh unit penguraian instruksi, dan dieksekusi oleh satu atau beberapa unit eksekusi di dalam CPU.



# Manajemen Memory

## Sejarah Memory

Istilah ini diciptakan oleh insinyur IBM George Scalise pada tahun 1956 ketika ia merancang RAMAC (Random Access Method of Accounting and Control), komputer komersial pertama dengan hard disk drive. Pada komputer awal, semua pemrosesan terjadi di memori utama karena tidak ada hard disk drive untuk penyimpanan jangka panjang program dan data.

Gagasan mengenai suatu jenis memori magnetik yang dapat diakses dengan cepat sudah ada sejak akhir tahun 1940-an. Perusahaan kecil seperti Engineering Research Associates dari St Paul, Minnesota, mengembangkan penyimpanan drum magnetik. Drum yang berputar ini dapat diandalkan tetapi lambat.



# Manajemen Memory

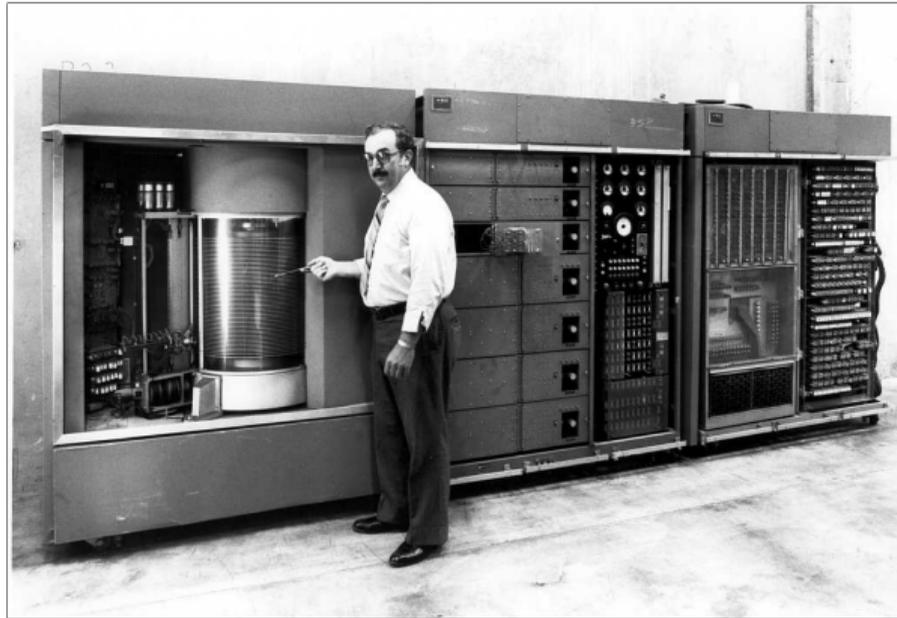
## Jenis Memory

Memory terdapat dua jenis:

- ▶ Main Memory/RAM
- ▶ Secondary Memory/Virtual Memory

# Manajemen Memory

## Gambar RAMAC





# Manajemen Memory

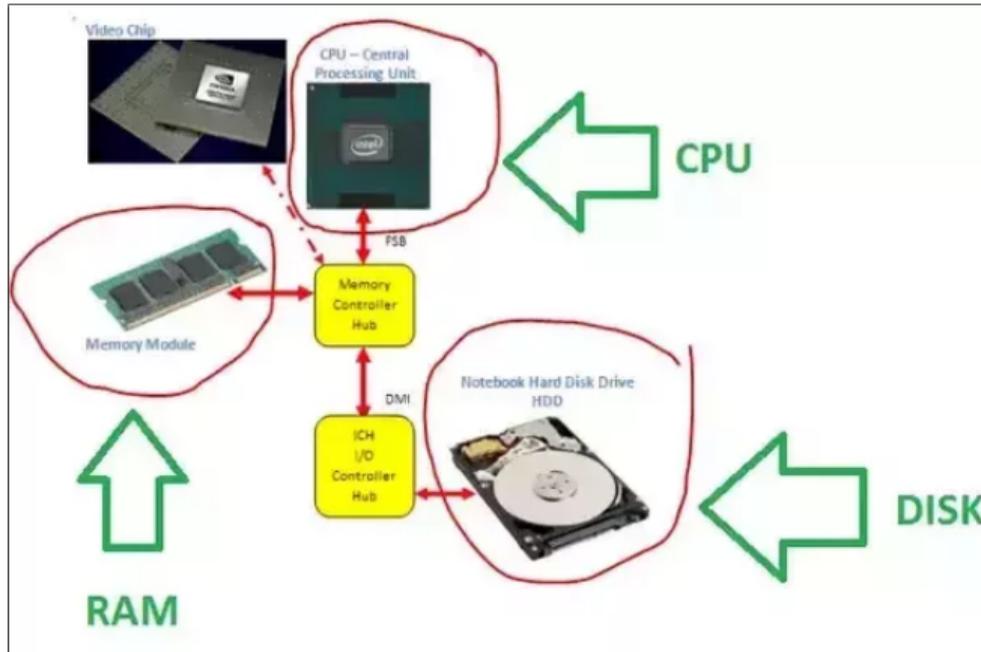
## Mengapa Memori dibutuhkan?

Komputer hanya dapat memanipulasi data yang ada di memori utama. Oleh karena itu, setiap program yang dijalankan dan setiap file yang diakses harus disalin dari perangkat penyimpanan ke dalam ruang memori utama. Jumlah ruang ini pada komputer sangat penting karena menentukan berapa banyak program yang dapat dieksekusi pada satu waktu dan berapa banyak data yang dapat tersedia untuk sebuah program.

**Disk** → **Memory/RAM** → **CPU**

# Manajemen Memory

## Gambar Loading Data





# Manajemen Memory

Lalu?

*Jika Memory digunakan untuk menyimpan data yang akan diproses sementara, lalu bagaimana sistem operasi memastikan memori dapat memenuhi kebutuhan masing-masing proses yang berjalan?*



# Manajemen Memory

## Manajemen Memori → Solusi

Manajemen memori adalah proses mengendalikan dan mengkoordinasikan memori utama komputer. Proses ini memastikan bahwa blok ruang memori dikelola dan dialokasikan dengan baik sehingga sistem operasi (OS), aplikasi, dan proses lain yang sedang berjalan memiliki memori yang dibutuhkan untuk menjalankan operasinya.

Sebagai bagian dari aktivitas ini, manajemen memori mempertimbangkan:

- ▶ Keterbatasan kapasitas perangkat memori
- ▶ Mengalokasikan ruang memori ketika tidak lagi diperlukan
- ▶ Mengoptimalkan penggunaan memori



# Manajemen Memory

Manajemen Memori → Solusi

Berikut ini adalah aspek-aspek kunci dari manajemen memori:

- ▶ Hirarki Memori
- ▶ Ruang Alamat
- ▶ Alokasi Memori
- ▶ Dealokasi Memori
- ▶ Perlindungan Memori



# Manajemen Memory

## Manajemen Memori → Hierarki Memori

Hirarki Memori: Sistem komputer modern biasanya memiliki hierarki memori yang terdiri dari beberapa tingkatan.

Setiap tingkat memiliki karakteristik yang berbeda dalam hal kecepatan, kapasitas, dan biaya. Sistem operasi mengelola pergerakan data di antara level-level ini untuk mengoptimalkan kinerja.

- ▶ Internal - Register dan cache prosesor.
- ▶ Utama - RAM sistem dan kartu pengontrol.
- ▶ Penyimpanan massal on-line - Penyimpanan sekunder.
- ▶ Penyimpanan massal off-line - Penyimpanan tersier dan Off-line.



# Manajemen Memory

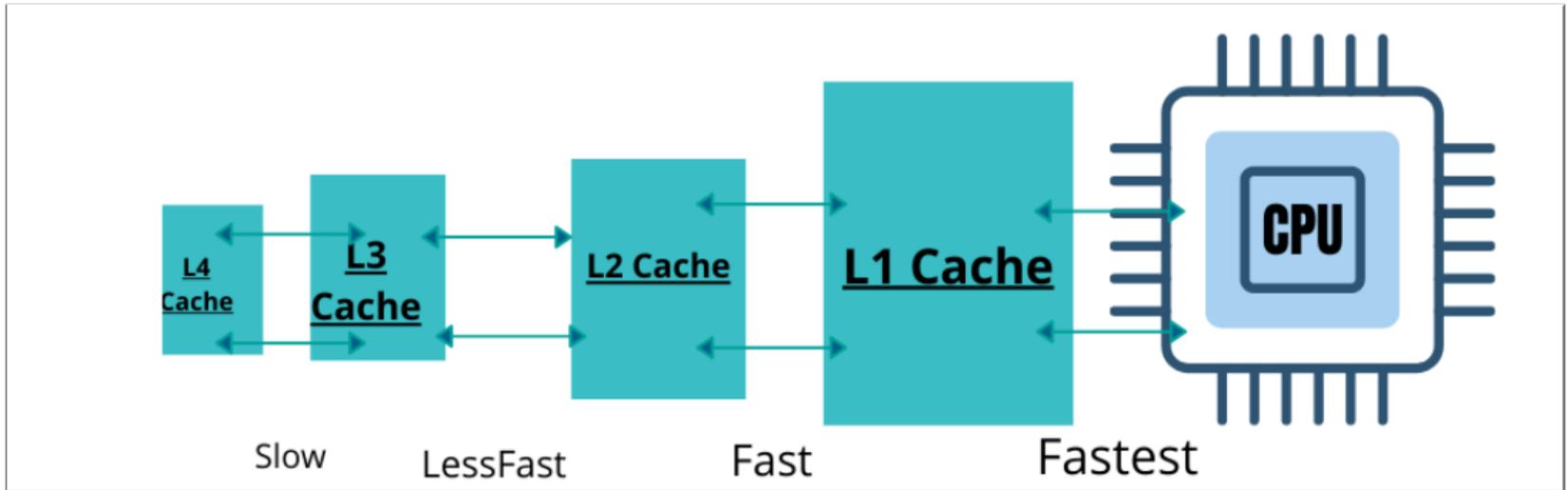
## Manajemen Memori → Hierarki Memori

Memori ini memiliki kecepatan yang tinggi namun kapasitas yang terbatas. Di gunakan oleh prosesor untuk meletakkan instruksi-instruksi.

- ▶ **Register:** unit memori kecil berkecepatan tinggi yang terintegrasi langsung ke dalam CPU. Register menyimpan data dan instruksi yang sedang diproses oleh CPU.
- ▶ **Cache:** memori kecil dan cepat yang terletak di antara CPU dan memori utama. Memori ini berfungsi sebagai penyangga untuk menyimpan data dan instruksi yang sering diakses, sehingga mengurangi waktu akses rata-rata.

# Manajemen Memory

Manajemen Memori → Hierarki Memori





# Manajemen Memory

## Manajemen Memori → Hierarki Memori

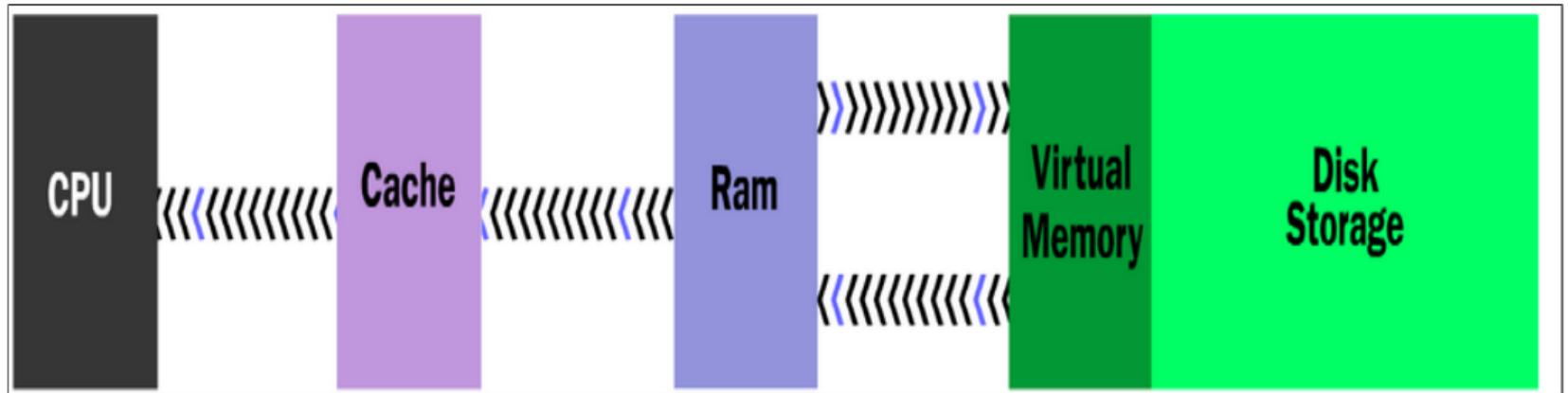
- ▶ **Memori Utama (RAM):** sering disebut sebagai Random Access Memory (RAM), memori ini menyimpan data dan instruksi yang secara aktif digunakan oleh CPU. Memori utama bersifat volatil, artinya isinya akan hilang ketika daya dimatikan.
- ▶ **Penyimpanan Sekunder:** perangkat penyimpanan yang tidak mudah hilang, seperti hard disk drive (HDD) dan solid-state drive (SSD).

### Informasi

Penyimpanan Sekunder dapat digunakan untuk membantu CPU dalam mengosongkan memori RAM.

# Manajemen Memory

Manajemen Memori → Hierarki Memori





# Manajemen Memory

Manajemen Memori → Ruang Alamat

Ruang alamat manajemen memori mengacu pada rentang alamat memori yang dapat diakses oleh suatu proses dalam sistem memori komputer. Ruang alamat memori dibagi menjadi beberapa segmen atau wilayah, masing-masing dengan tujuan tertentu atau serangkaian izin akses.

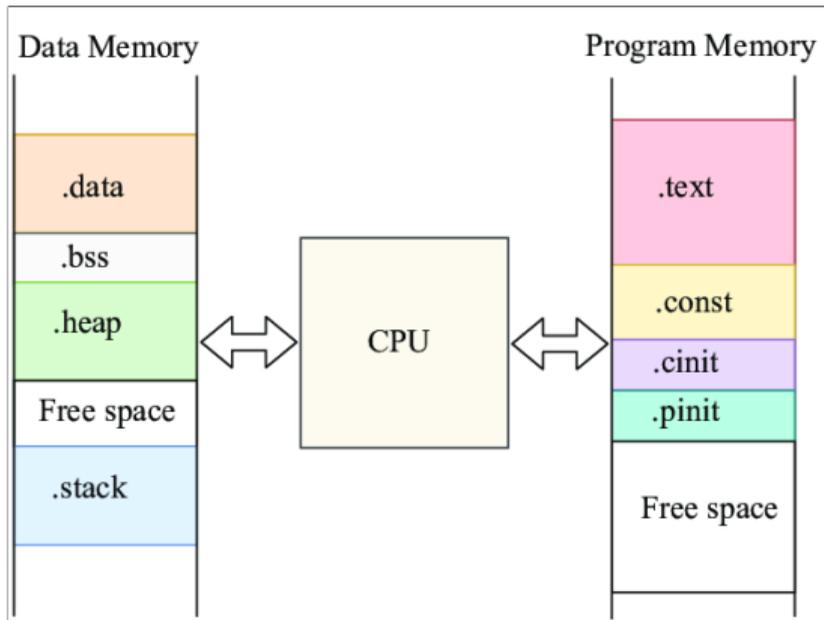
Sistem Operasi membagi ini menjadi 4 bagian

- ▶ Segmen Teks/Kode
- ▶ Segmen Data
- ▶ Segmen BSS
- ▶ Segmen Tumpukan/Stack



# Manajemen Memory

Manajemen Memori → Ruang Alamat





# Manajemen Memory

Manajemen Memori → Ruang Alamat

**Segmen Teks** (juga dikenal sebagai Segmen Kode): Segmen ini berisi kode program yang dapat dieksekusi. Segmen teks hanya dapat dibaca, yang berarti program tidak dapat memodifikasi instruksi yang tersimpan dalam segmen ini.

- ▶ Segmen teks berisi instruksi yang dapat dieksekusi dari program C.
- ▶ Termasuk semua fungsi yang membentuk program
- ▶ Segmen teks dapat dibagi sehingga hanya satu salinan yang perlu ada di memori untuk program eksekusi yang berbeda
- ▶ Biasanya, segmen teks bersifat read-only, untuk mencegah sebuah program secara tidak sengaja mengubah instruksinya.



# Manajemen Memory

Manajemen Memori → Ruang Alamat

**Segmen Data** berisi data statis yang diinisialisasi dan variabel global yang digunakan oleh program. Segmen ini dapat ditulis, yang berarti bahwa program dapat memodifikasi data yang disimpan dalam segmen ini selama waktu proses.

**Segmen BSS** berisi data statis yang belum diinisialisasi dan variabel global yang digunakan oleh program. Segmen ini juga dapat ditulis, dan sistem operasi menginisialisasi data di segmen ini menjadi nol.

**Segmen Tumpukan** digunakan untuk menyimpan variabel lokal dan frame pemanggilan fungsi.



# Manajemen Memory

Manajemen Memori → Alokasi dan Dealokasi Memori

**Alokasi Memori** dilakukan ketika sebuah proses dibuat atau meminta memori, sistem operasi perlu mengalokasikan memori untuk proses tersebut. Dalam melakukan alokasi, Sistem akan menerapkan dua hal:

- ▶ **Contiguous Allocation:** Memori dibagi menjadi partisi berukuran tetap, dan setiap proses dialokasikan blok memori yang bersebelahan.
- ▶ **Non-Contiguous Allocation:** Membagi memori ke dalam blok-blok ukuran tetap yang disebut pages.



# Manajemen Memory

Manajemen Memori → Alokasi dan Dealokasi Memori

Ketika sebuah proses selesai atau secara eksplisit melepaskan memori, sistem operasi perlu mengalokasikan ulang sumber daya memori. Proses ini menandai memori yang bersangkutan sebagai bebas dan memperbarui struktur data yang digunakan untuk manajemen memori.

Terdapat berbagai cara yang dapat digunakan untuk melakukan manajemen alokasi memori ini.



# Manajemen Memory

Manajemen Memori → Alokasi dan Dealokasi Memori

Teknik dealokasi yang sering digunakan adalah:

- ▶ **Explicit Deallocation** : Secara langsung meminta sistem untuk mengosongkan. Menggunakan metode **free()** atau **delete()**
- ▶ **Garbage Collection** : Menggunakan teknik yang sudah disediakan oleh program secara otomatis
- ▶ **Stack Deallocation** : Memori yang dialokasikan pada stack secara otomatis dialokasikan ulang ketika fungsi atau cakupan blok berakhir.
- ▶ **OOM Killer** : Mengelola alokasi memori dalam situasi di mana sistem kehabisan memori yang tersedia. Ketika memori yang tersedia habis dan sistem tidak dapat mengalokasikan lebih banyak memori untuk proses, OOM Killer akan bekerja.



# Manajemen Memory

Manajemen Memori → Perlindungan Memori

Perlindungan memori merupakan aspek penting dari manajemen memori. Sistem operasi memberlakukan kontrol akses ke wilayah memori untuk mencegah akses atau modifikasi yang tidak sah. Sistem ini menggunakan teknik seperti memori hanya-baca (ROM), izin akses, dan segmentasi memori untuk memastikan integritas dan keamanan sistem.



# Manajemen Memory

## *Last Question*

*Jika sebuah sistem operasi mengalokasikan proses mencapai titik kapasitas tertentu dari Memory Utama, apa yang dilakukan oleh Sistem Operasi untuk memastikan Memory Utama tetap tidak penuh?*

