# 5 Chức năng cốt lõi của Hệ điều hành

## 1. Quản lý bộ nhớ (Memory Management)

Mục tiêu: Phân bổ, bảo vệ và tối ưu dung lượng RAM cho các tiến trình; hỗ trợ bộ nhớ ảo.

Cách hoạt động:

- Ánh xạ địa chỉ ảo sang địa chỉ vật lý bằng page table/segment table.

- Sử dụng kỹ thuật phân trang (paging) và phân đoạn (segmentation).

- Dùng bộ nhớ ảo (virtual memory) và swap khi RAM thiếu.

Ví dụ thực tế: Khi mở nhiều tab trình duyệt trên Windows hoặc Linux, hệ điều hành chuyển một số trang ít dùng vào ổ đĩa (swap/pagefile) để giải phóng RAM.

## 2. Quản lý thiết bị nhập/xuất (I/O & Device Management)

Mục tiêu: Điều phối giao tiếp giữa CPU và thiết bị ngoại vi.

Cách hoạt động:

- Sử dụng device driver để kết nối phần cứng.

- Dùng ngắt (interrupt) để nhận tín hiệu từ thiết bị.

- Sử dụng DMA để truyền dữ liệu trực tiếp vào RAM, giảm tải CPU.

Ví dụ thực tế: Khi in tài liệu, hệ điều hành dùng driver máy in và bộ đệm để gửi dữ liệu, máy in báo ngắt khi hoàn tất.

## 3. Cung cấp giao diện người dùng (User Interface)

Mục tiêu: Cho phép người dùng tương tác với hệ điều hành thông qua GUI hoặc CLI.

Cách hoạt động:

- CLI (dòng lệnh): Shell nhận và phân tích lệnh, gọi API hệ thống.

- GUI (đồ họa): Window manager xử lý sự kiện chuột/phím và hiển thị cửa sổ.

Ví dụ thực tế: Mở File Explorer (Windows) là GUI; gõ 'dir' hoặc 'ls' trong terminal là CLI.

## 4. Quản lý hệ thống tệp (File System Management)

Mục tiêu: Tổ chức và quản lý lưu trữ dữ liệu trên ổ đĩa.

Cách hoạt động:

- Dùng cấu trúc dữ liệu như inode, MFT hoặc FAT để lưu metadata.

- Dùng journaling để bảo toàn dữ liệu khi xảy ra sự cố.

- Quản lý quyền truy cập và mount phân vùng.

Ví dụ thực tế: Windows dùng NTFS (journaling, quyền truy cập), Linux dùng ext4. Khi lưu file, hệ điều hành ghi dữ liệu vào block và cập nhật bảng metadata.

## 5. Quản lý tiến trình (Process Management)

Mục tiêu: Tạo, lập lịch, đồng bộ và tiêu hủy tiến trình; phân chia CPU.

Cách hoạt động:

- Mỗi tiến trình có PCB chứa trạng thái.

- OS dùng thuật toán lập lịch như round-robin, priority scheduling.

- Hỗ trợ cơ chế đồng bộ và giao tiếp liên tiến trình (IPC).

Ví dụ thực tế: Khi chạy trình duyệt, OS tạo nhiều tiến trình (tab, plugin). Scheduler phân chia CPU; nếu tab treo, bạn có thể kết thúc tiến trình qua Task Manager.