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

### Giải thích:

- Hệ điều hành chịu trách nhiệm phân phối và giám sát việc sử dụng bộ nhớ RAM.

- Khi một chương trình hoặc tiến trình được chạy, OS cấp phát một phần bộ nhớ để nó hoạt động.

- Sau khi tiến trình kết thúc, OS thu hồi lại vùng nhớ để tái sử dụng.

- Ngoài RAM, hệ điều hành còn quản lý **bộ nhớ ảo (virtual memory)**, cho phép sử dụng ổ cứng như một phần mở rộng của RAM.

### Ví dụ thực tế:

- Khi mở nhiều tab trong trình duyệt Chrome, mỗi tab được coi như một tiến trình. Hệ điều hành sẽ cấp phát một vùng nhớ riêng cho từng tab để chúng không ảnh hưởng lẫn nhau.

- Trên Windows, khi RAM đầy, hệ điều hành sẽ dùng **file hoán trang (pagefile.sys)** trên ổ cứng làm bộ nhớ ảo.

### Tóm tắt:

Quản lý bộ nhớ giúp đảm bảo phân phối RAM công bằng và hiệu quả cho các ứng dụng, tránh tràn bộ nhớ, đồng thời tận dụng bộ nhớ ảo để mở rộng dung lượng.

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

### Giải thích:

- OS là cầu nối giữa phần mềm và phần cứng nhập/xuất như bàn phím, chuột, màn hình, máy in, ổ cứng.

- Nó sử dụng **trình điều khiển (driver)** để giao tiếp với từng loại thiết bị.

- Hệ điều hành tổ chức việc truy cập, chia sẻ và ngăn xung đột khi nhiều tiến trình cùng muốn sử dụng một thiết bị.

### Ví dụ thực tế:

Khi bạn nhấn **Ctrl + P** để in tài liệu trong Word, OS sẽ gửi lệnh đến máy in qua driver máy in.

Khi bạn cắm USB, Windows/Linux sẽ tự động nhận diện và gán ký hiệu ổ đĩa (E:, F:).

### Tóm tắt:

Quản lý I/O cho phép người dùng và chương trình truy cập thiết bị phần cứng dễ dàng, đảm bảo các tiến trình có thể sử dụng thiết bị mà không cần quan tâm đến chi tiết kỹ thuật.

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

### Giải thích:

- Giao diện người dùng là cách để người dùng tương tác với hệ thống.

- Có hai loại chính:

**CLI (Command-Line Interface)**: giao diện dòng lệnh (như Terminal trong Linux, CMD trong Windows).

**GUI (Graphical User Interface)**: giao diện đồ họa với cửa sổ, biểu tượng, menu (Windows, macOS).

### Ví dụ thực tế:

- Khi bạn nhấp chuột vào biểu tượng **Recycle Bin** để mở, đó là bạn đang dùng GUI.

- Khi lập trình viên gõ lệnh ls trong Linux Terminal để liệt kê file, đó là CLI.

### Tóm tắt:

Cung cấp giao diện người dùng giúp mọi người dễ dàng điều khiển máy tính, từ người bình thường (GUI) đến lập trình viên hay quản trị viên (CLI).

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

### Giải thích:

- Hệ điều hành tổ chức và lưu trữ dữ liệu trong các tập tin (file) và thư mục (folder).

- Nó đảm bảo người dùng có thể tạo, đọc, ghi, chỉnh sửa, xóa file.

- OS cũng quản lý quyền truy cập (ai được đọc/ghi/xóa).

- Các hệ điều hành có thể dùng các hệ thống tệp khác nhau: NTFS (Windows), ext4 (Linux), APFS (macOS).

### Ví dụ thực tế:

- Trên Windows, khi bạn tạo một file Word và lưu vào ổ C:, OS sẽ ghi file đó vào hệ thống NTFS.

- Trên Linux, quyền file được phân biệt thành **read, write, execute** cho owner, group, others.

### Tóm tắt:

Quản lý hệ thống tệp giúp dữ liệu được lưu trữ có tổ chức, an toàn và dễ truy cập.

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

### Giải thích:

- Tiến trình (process) là một chương trình đang chạy trong bộ nhớ.

- Hệ điều hành quyết định tiến trình nào chạy, khi nào chạy, bao nhiêu CPU được cấp phát.

- OS dùng **bộ lập lịch (scheduler)** để phân bổ CPU cho các tiến trình.

- OS cũng hỗ trợ **đa nhiệm (multitasking)**, giúp nhiều tiến trình chạy song song.

### Ví dụ thực tế:

- Khi bạn nghe nhạc trên Spotify, đồng thời mở Word và Chrome, OS sẽ chia CPU để cả ba tiến trình hoạt động.

- Trên Task Manager (Windows), bạn có thể thấy danh sách các tiến trình đang chạy, như explorer.exe, chrome.exe.

### Tóm tắt:

Quản lý tiến trình giúp hệ điều hành điều phối tài nguyên CPU, đảm bảo các ứng dụng hoạt động mượt mà và có thể chạy đồng thời.

**Kết luận:**  
- 5 chức năng cốt lõi của hệ điều hành (quản lý bộ nhớ, quản lý I/O, cung cấp giao diện người dùng, quản lý hệ thống tệp, quản lý tiến trình) là nền tảng giúp máy tính hoạt động hiệu quả, cho phép phần mềm và phần cứng phối hợp nhịp nhàng để phục vụ người dùng.