**>>> Bài 3:**

1. Two-state process model

Not Running - Running

=> Chỉ áp dụng với tiến trình đã sẵn sàng

2. Five-state process model

New - Ready - Running - Blocked - Exit

=> Tiến trình cần dừng lại để chờ hoạt động I/O

3. Suspend State

New - Ready - Running - Blocked - Suspend - Exit

=> Bộ xử lý nhanh hơn hoạt động I/O rất nhiều, khi tiến trình chờ I/O thì bộ xử lý rất nhàn rỗi, các tiến trình khác vẫn phải đợi, cần nhường bộ xử lý cho các tiến trình khác

=> Tối ưu hoạt động CPU

**>>> Bài 1**

Windown API

Hầu hết các hàm API được chứa trong các file DLL (Dynamic Link Library – thư viện liên kết động) chứa trong thư mục hệ thống của Windows (C:\Windows\System32). Các hàm API được biên dịch độc lập với các ứng dụng, khi chương trình ứng dụng được biên dịch và thực thi, chúng gọi tới các hàm API, và các chương trình được link tham chiếu tới file .dll (mapping).

Một số thư viện:

- Kernel32.dll là file DLL thao tác với lõi Windows, nó chứa các hàm API ở cấp độ thấp (thường được sử dụng) quản lý bộ nhớ (memory) và tác vụ (task).

- User32.dll là hàm cơ bản để quản lý cửa sổ, nhập dữ liệu, chứa các hàm xử lý menu, định giờ, truyền tin,…

- Gdi32.dll quản lý giao diện đồ họa.

+ Advapi32.dll Hỗ trợ rất nhiều hàm API (ở mức độ chuyên sâu cao-Advanced), liên quan tới security, registry

+ Comdlg32.dll Thư viện API common dialog

+ Mpr.dll Thư viện Multiple Provider Router

+ Netapi32.dll Thư viện 32 bit-Network

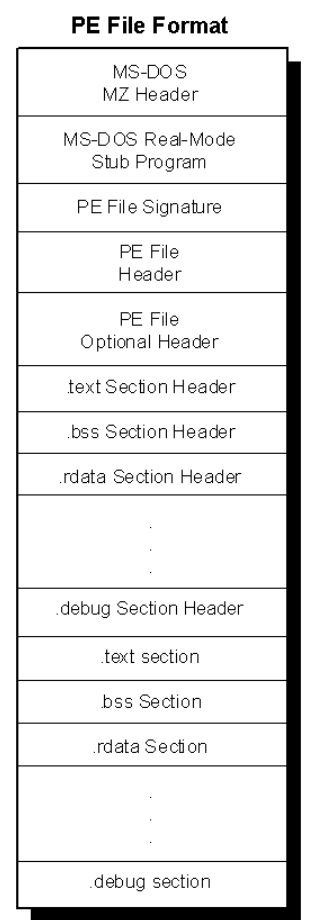
+ Shell32.dll Thư viện 32 bit shell

+ Winmm.dll Thư viện Windows multimedia

+ Winspool.drv Nơi chứa các hàm API làm việc với máy in, Printer interface

https://docs.microsoft.com/en-us/windows/win32/apiindex/windows-api-list

**>>> Bài 2**



EXE file:

- MS-DOS Header: chứa thông tin vị trí, kích thước của mã và vùng dữ liệu, tệp dành cho hệ điều hành, kích thước ngăn xếp ban đầu và các thông tin quan trọng khác

- Real-Mode Stub Program: một chương trình thực tế được chạy bởi MS-DOS khi được tải

- File Header and Signature: tiêu đề và chữ ký tệp

- Optional Header: Chứa các thông như kích thước ngăn xếp ban đầu, vị trí điểm nhập chương trình, địa chỉ cơ sở ưa thích, phiên bản hệ điều hành, thông tin căn chỉnh phần, …

- Executable code section, .text: phần mã thực thi

- Data sections, .bss, .rdata, .data:

.bss biểu thị dữ liệu chưa được khởi tạo cho ứng dụng, bao gồm tất cả các biến được khai báo là tĩnh trong một mô-đun chức năng hoặc nguồn.

.rdata biểu thị dữ liệu chỉ đọc, chẳng hạn như chuỗi ký tự, hằng và thông tin thư mục gỡ lỗi.

.data lưu trữ tất cả các biến khác (ngoại trừ biến tự động, xuất hiện trên ngăn xếp). Về cơ bản, đây là các biến toàn cầu ứng dụng hoặc mô-đun.

- Resources section, .rsrc: chứa thông tin tài nguyên

- Export data section, .edata: chứa dữ liệu xuất cho ứng dụng hoặc DLL

- Import data section, .idata: nhập dữ liệu, bao gồm thư mục nhập và bảng tên địa chỉ nhập

- Debug information section, .debug: Thông tin gỡ lỗi ban đầu

<https://blog.kowalczyk.info/articles/pefileformat.html>

<https://docs.microsoft.com/en-us/previous-versions/ms809762(v=msdn.10)?redirectedfrom=MSDN>

<https://www.quora.com/What-is-the-file-structure-of-an-exe-file-What-is-the-memory-location-of-its-starting-address>