Chapter 2: OS Structures

19/08/19

OS Services :

- Users
  - UI
  - Program Execution
  - I/O operation
  - File system manipulation
  - Communication (shared memory / message passing)
  - Error detection
- Programs
  - Resource allocation
  - Logging
  - Protection / Security

User / OS Interface :

- CLI : shells like BASH, ZSH, KSH, C-Shell
- GUI : Aqua for MacOS, KDE, GNOME

System Calls :

- Use POSIX / Windows API
- interface to use services provided by OS
- params for syscalls passed to OS in three ways — registers, storing in block / table in memory w/ address being passed or using a stack

- Types
    - Process control
    - file management
    - Device management
    - Information maintenance
    - Communication
    - Protection

System Services:

- file management
- status information
- program load and exec
- background services
- file modification
- programming lang support
- communication

OS Structure:

- Monolithic Structure
    - no structure
    - single address space
    - single static binary file
    - difficult to implement/extend
    - very little system call interface overhead
    - fast intra kernel communication
- Layered Approach
    - loosely coupled system
    - simplicity of construction, debugging

- Microkernel
    - all non essential components removed from kernel
    - implemented as system programs
    - easy extension of OS
    - performance may suffer due to system function overhead
- Loadable Kernel Modules (LKM)
    - core components + linkables services as modules
    - similar to layered