# Log Book Week 2

Jose Devian Hibono
1706039603
System Programming - C

---

## System Calls

In order for programs to interact with the operating system, we will need a system call. The system call acts like a bridge between user mode and the operating system. System call provides the services of the operating system to the user programs via Application Program Interface (API).

## User Mode, Kernel Mode, and System Call

As previously discussed, there's a massive difference of privilege between user mode and kernel mode. So in order of the user mode to access the command to the OS, they will need a system call.

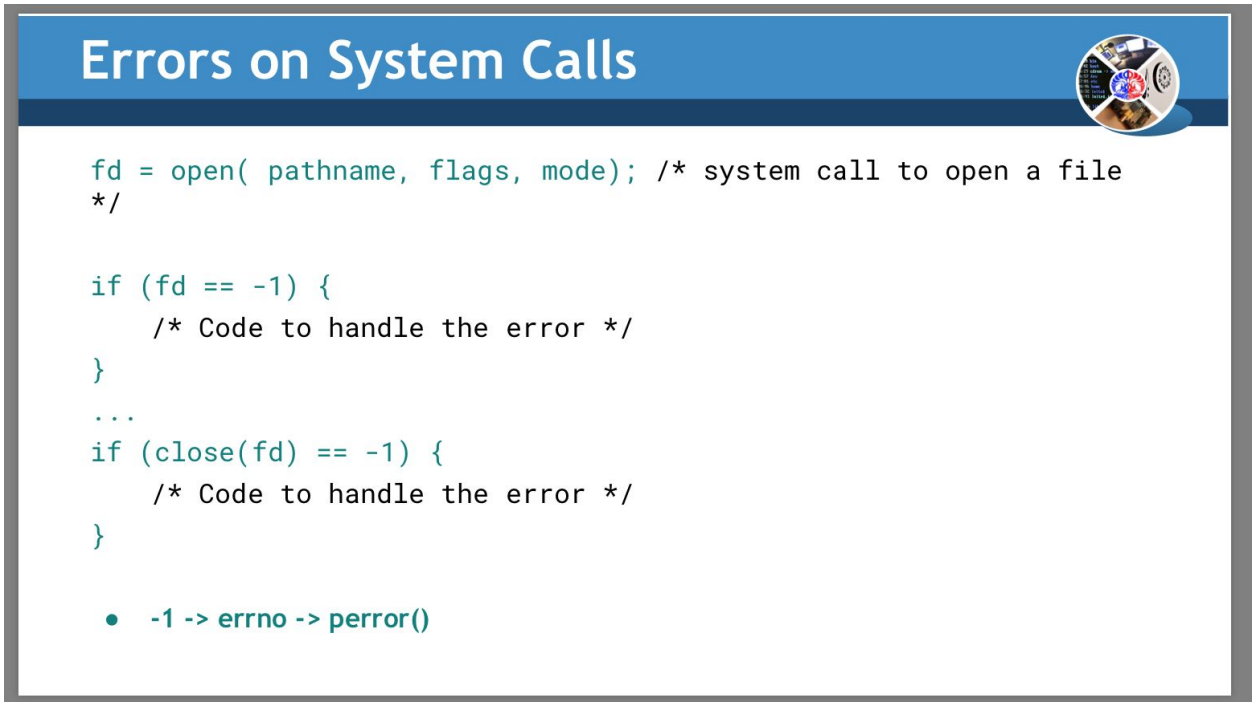
## Execution of System Call

1. Application Program runs a wrapper function
2. The wrapper function will do something to trigger the execution of a system call.
3. The system will switch from User Mode to Kernel Mode in order to execute the system call.
4. After switched, the command in the wrapper function will be handled by Trap Handler before the execution of the system call.
5. The trap handler will execute the system call.
6. The call will be handled by the System Call Service Routine.
7. The system call will be processed in this service, then it returns a value.
8. Then the trap handler will return the value to the wrapper function.
9. After that, it will switch back to user mode. Then the wrapper function will return the previous value to the application program.

## Library Functions

The functions which are a part of the standard C library are known as Library functions. Library functions are different from system functions. There are two kinds of Library Functions: (1) Functions that do not call any system call and (2) Functions that make a system call.

## System Call Errors




Source: Course's Slide

System Call Errors is collected in the file `errno.h`

Several examples of errno:

i. `errno(2)`: No such file or directory, when you're trying to access a false pathname. It might be a typo or it doesn't exist.
ii. `errno(5)`: Input/output error, there's a problem in the input or output.
iii. `errno(12)`: Cannot allocate memory, memory is full or there's an error in the memory.