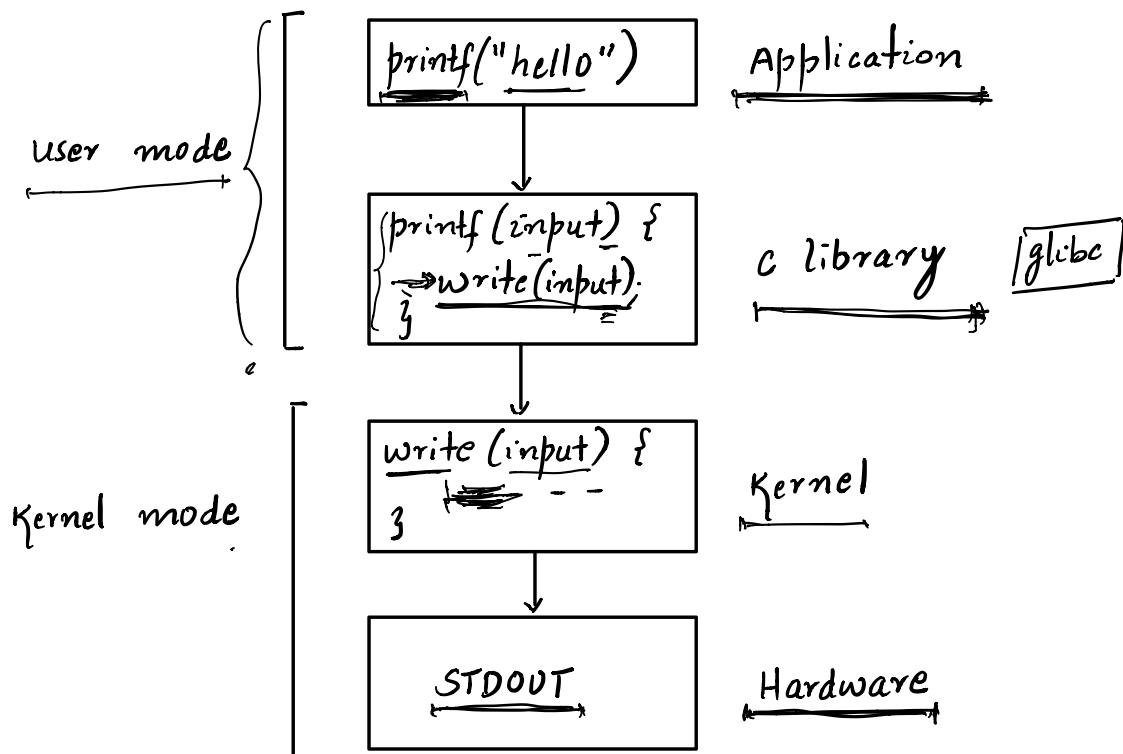


SYSTEM CALLS IN OPERATING SYSTEM

- User applications cannot directly communicate with the hardware devices.
- Then, how will an application write to a file that is stored on the disk(hardware)?
- User applications can take help of the system calls and communicate with the hardware devices.
- Operating system acts as an interface between user programs and hardware devices.
- The interface that an operating system provides to user applications is system call interface.



API AND SYSTEM CALLS

- Application programming interface
- Set of functions exposed by any library - printf
- System calls are also functions exposed by C library
- So, system calls also is an API
- API - system call - write () ✓
- API - uses system calls - printf()
- API - doesn't uses system calls - strlen()

→ qsort (- -)
glibc

→ write ()

→ read ()

printf () ←

write ()

?

Examples of system calls

- write(), read(), open(), close() — file manipulation
- fork(), execve(), exit(), wait4() — Process manage
- connect(), send(), recv(), socket() — Socket/network programming
- mkdir(), chown(), chmod() — Files

