

Date 25-9-21 :

library functions vs system calls

library functions

- library functions are supported by compiler
- another name is API
(application programming interface)

Application Programming

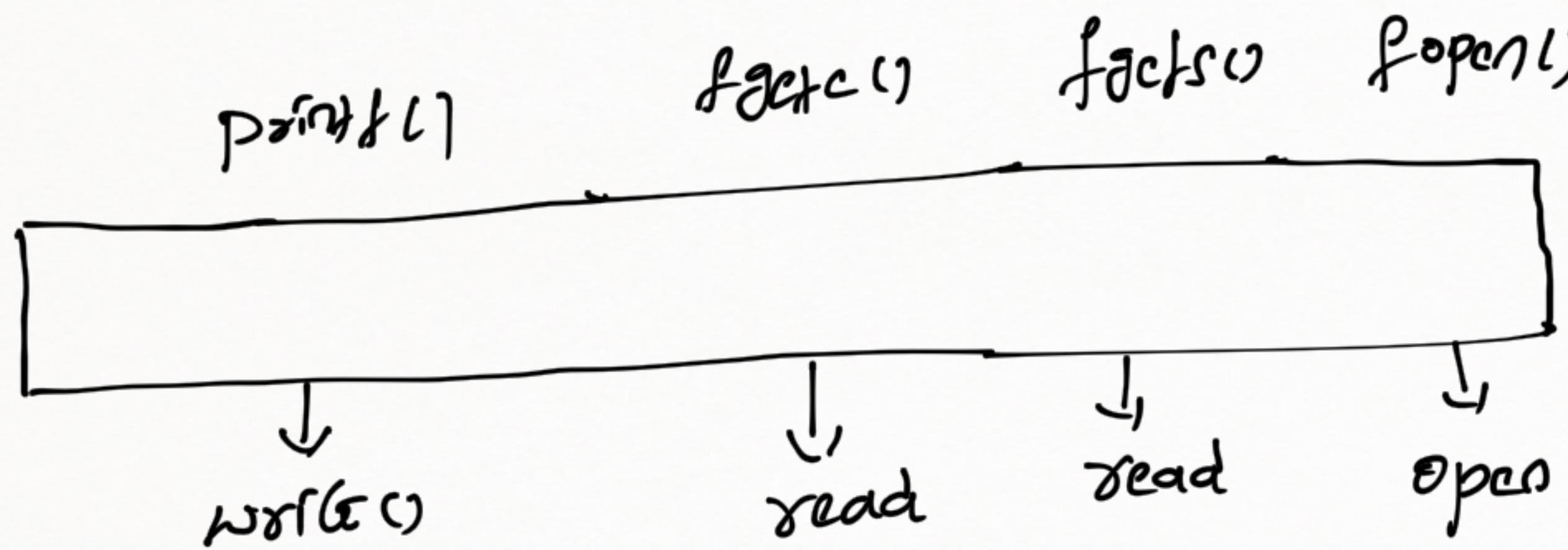
system calls

- system calls are supported by OS
- another name is SCI
(System call interface)

System Programming

→ Library functions are programmer friendly and specific to task

fgetc(), fgetsc()



→ library functions are porttask

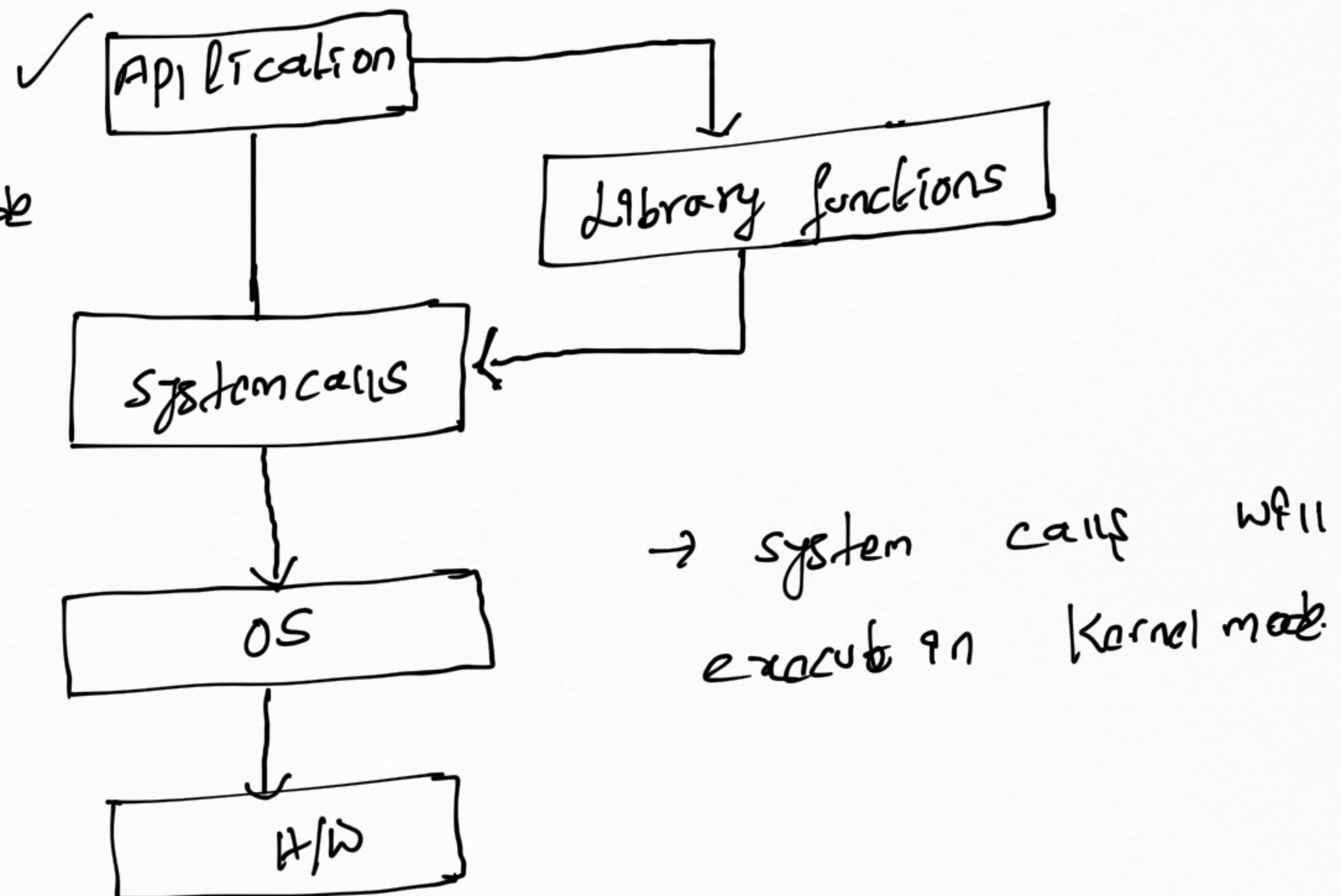
→ system calls are generic in nature

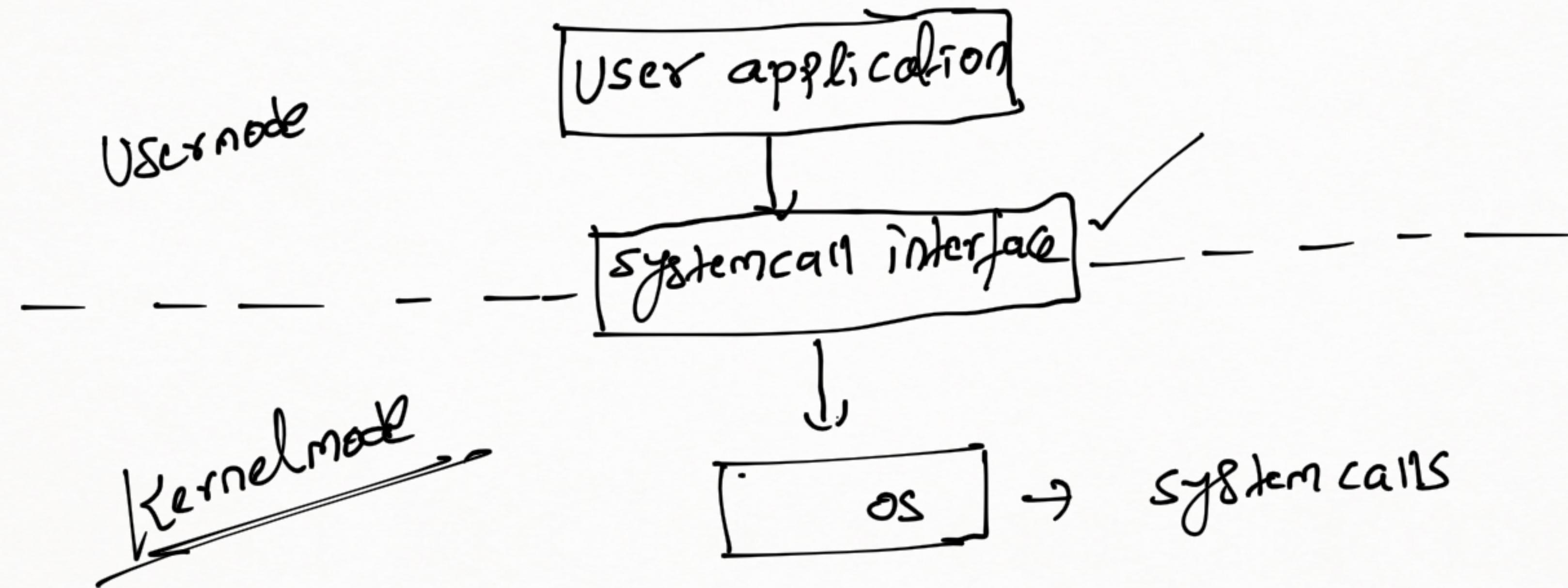
read(), writ()

→ system calls are not portable (cos to os system call interface different).

✓
system calls are
interface b/w user mode
and kernel mode

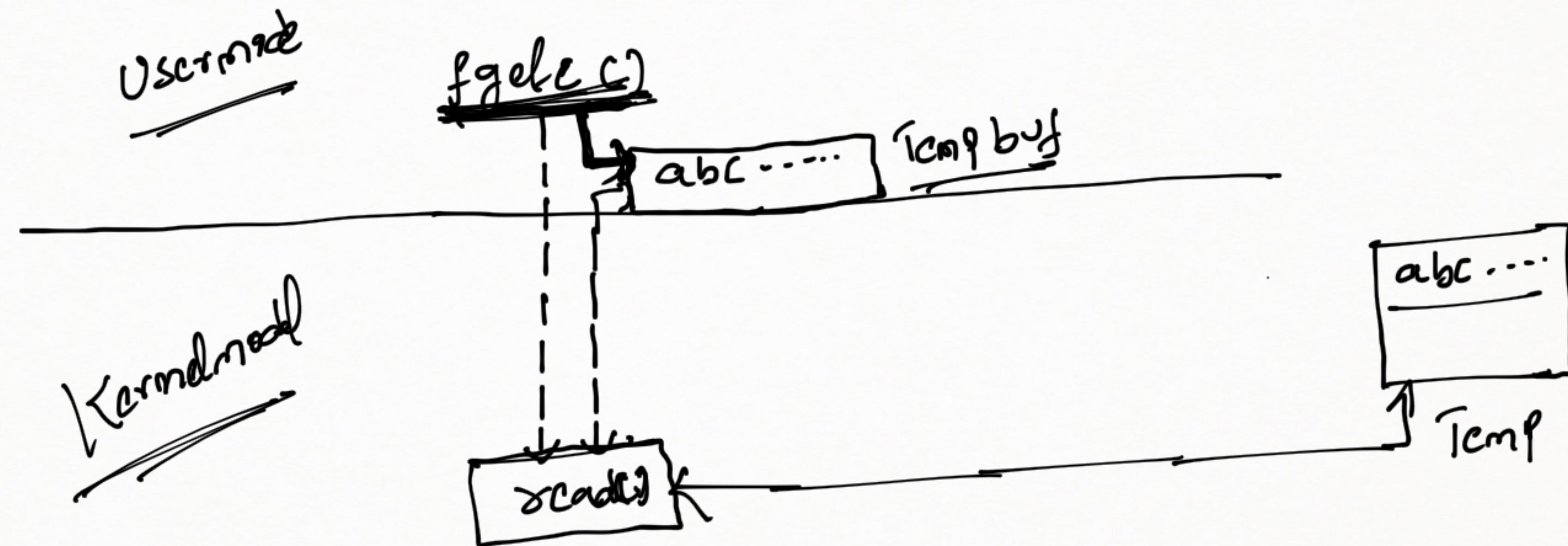
→ Library functions w/fil
execute in user mode





→ If application needs kernel services directly from system calls usage deprecated

`open()`
`read()`
`write()`
`close()`
`fctrl()`
`dup()`
`dup2()`



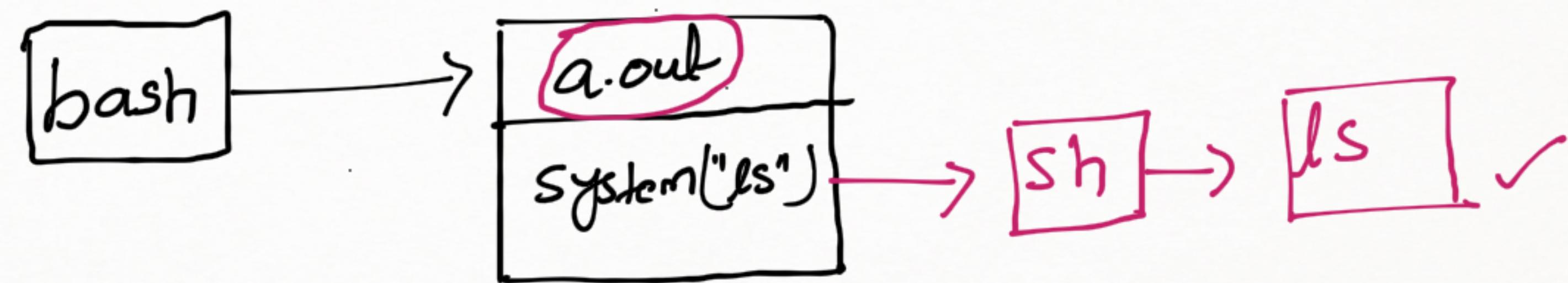
system function

→ Executing shell command

hi ✓

by ✓

```
main()
{
    printf("hi..ln")
    system("ls");
    printf("byln")
}
```



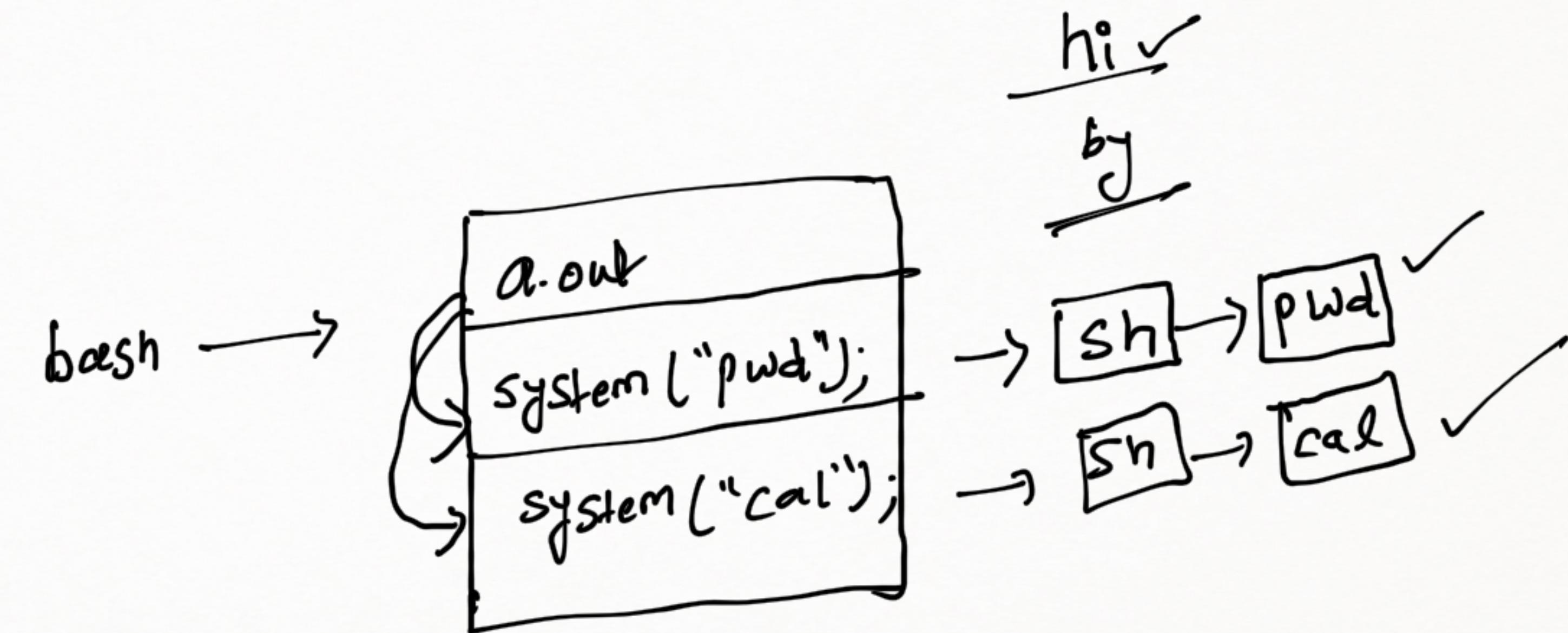
Bash → a.out → sh → ls

system functions calls the sh shell

```

main()
{
    printf("hi--\n");
    system ("pwd");
    system ("cal");
    printf("by--\n");
}

```



bash → a.out → sh → pwd

bash → a.out → sh → cal