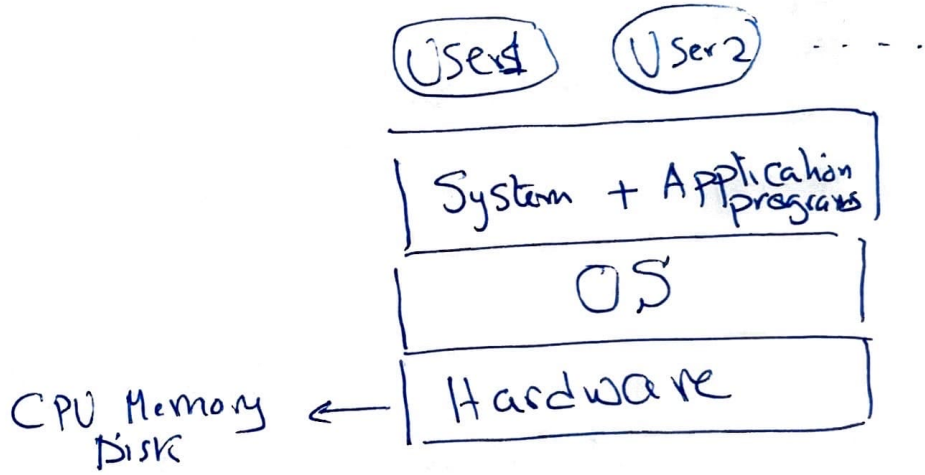
What is an OS?

Program that manages the computer resources (hardware).

write()

It acts as an interface to the hardware for application programs.



Multiple Views of an OS

User's View

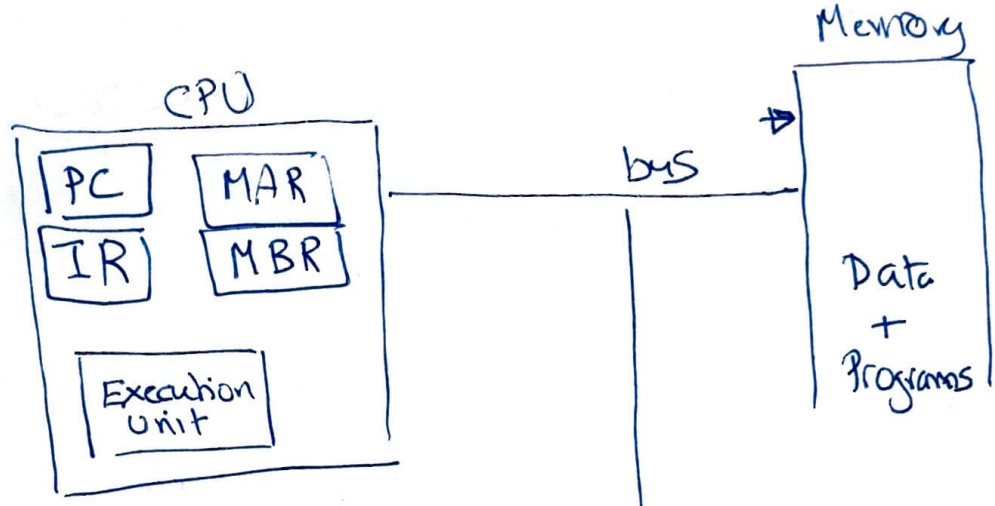
PC, Ease of use.

System's View

OS as a resource allocator

* Fairness
* Efficiency
* load balancing

Computer System Organization



PC: Program Counter

IR: Instruction Register

MAR: Memory Address Register
MBR: " Buffer Register.

PSW: Program Status Word.

I/O Input/Output

I/O module.

Operation

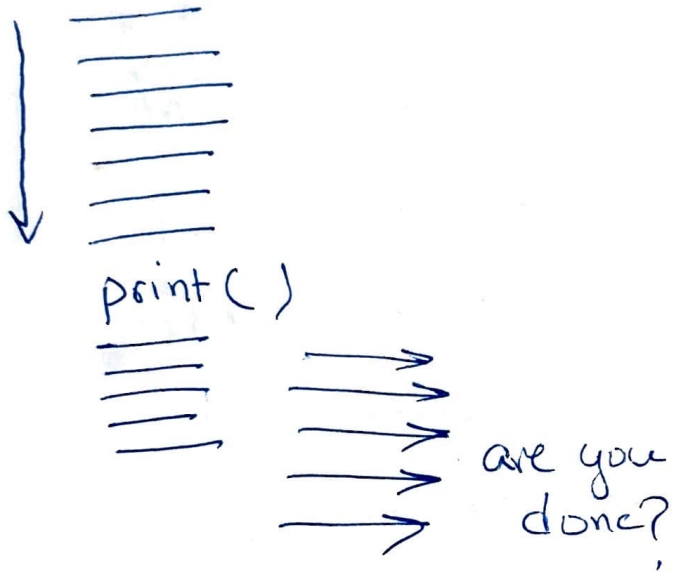
when a machine is booted:

* an initial program (stored in ROM or EEROM) executes.
* Initializes the System.
* Loads part of the OS into memory
* OS executes its first process (init).
* Waits for events to happen.

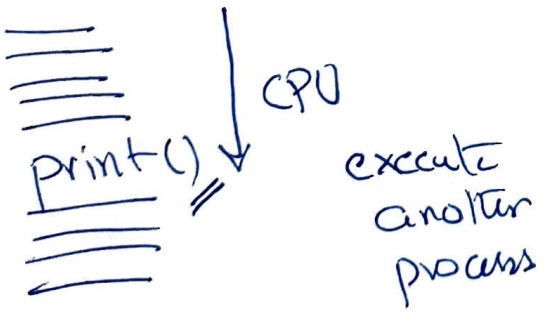
Events are signalled by interrupts

Interrupt: a mechanism to interrupt the CPU execution of instructions.

Program



With interrupts

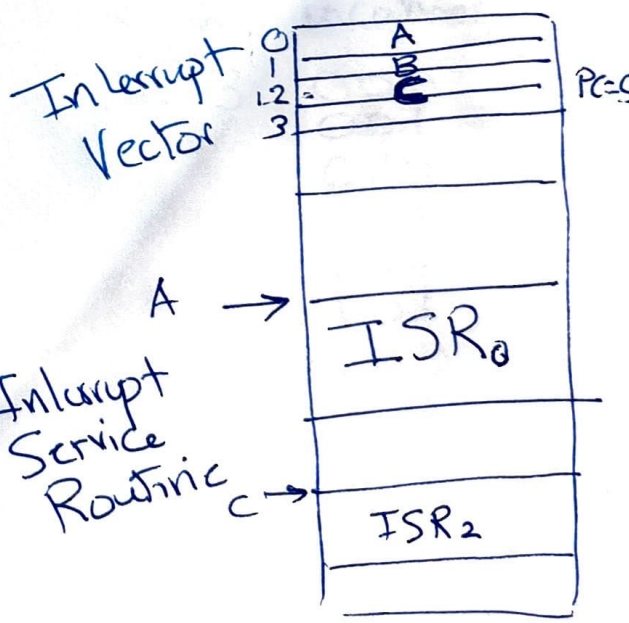


Why Interrupts?

Keep the CPU utilization high given that I/O devices are MUCH slower than the CPU.

Interrupt Processing

1. An interrupt occurs
2. CPU finishes executing the current instruction.
3. Acknowledge the interrupt.
4. CPU pushes the PC + PSW on the stack.
5. Save the rest of the process state (registers).
6. CPU loads PC with the address of the ISR.
7. Execute the ISR.
8. Restore the process state.
9. Pop PC + PSW from stack.



Storage

CPU loads instructions from memory (RAM).

2^n locations.

3
000
001
010
011
100
101
110
111

Storage hierarchy

- Registers
- Cache
- Main memory
- Electronic disks
- Magnetic disks
- Optical disks
- Magnetic Tapes

I/O

Wide variety of devices that the OS must handle.

Device drivers: provide an interface to the OS to interact with the device..