LEC-2: Types of OS

**OS goals –**

- Maximum CPU utilization

- Less process starvation

- Higher priority job execution

Types of operating systems –
-     Single process operating system   [MS DOS, 1981]
-     Batch-processing operating system   [ATLAS, Manchester Univ., late 1950s – early 1960s]
-     Multiprogramming operating system   [THE, Dijkstra, early 1960s]
-     Multitasking operating system   [CTSS, MIT, early 1960s]
-     Multi-processing operating system   [Windows NT]
-     Distributed system   [LOCUS]
-     Real time OS   [ATCS]

```
Types of OS

  → Single process operating system

  → Batch-processing operating system

  → Multi-programming operating system

  → Multitasking operating system

  → Multi-processing operating system

  → Distributed operating system

  → Real Time OS
```
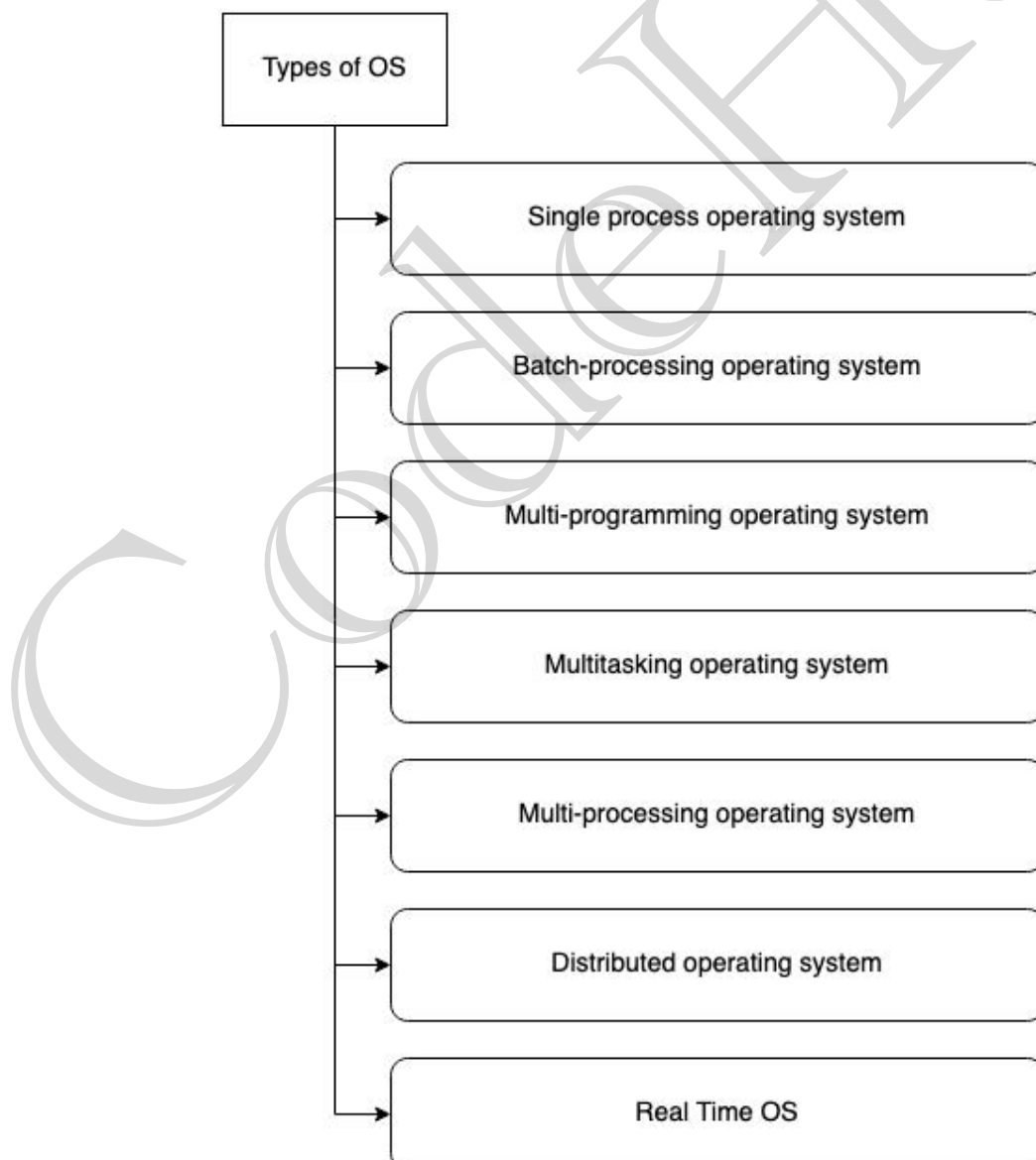
Single process OS, only 1 process executes at a time from the ready queue. [Oldest]
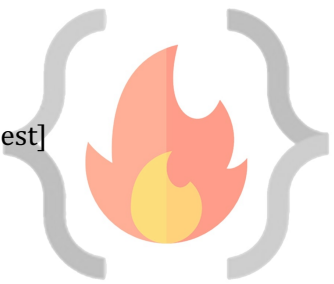
Batch-processing OS,
1. Firstly, user prepares his job using punch cards.
2. Then, he submits the job to the computer operator.
3. Operator collects the jobs from different users and sort the jobs into batches with similar needs.
4. Then, operator submits the batches to the processor one by one.
5. All the jobs of one batch are executed together.

- Priorities cannot be set, if a job comes with some higher priority.
- May lead to starvation. (A batch may take more time to complete)
- CPU may become idle in case of I/O operations.
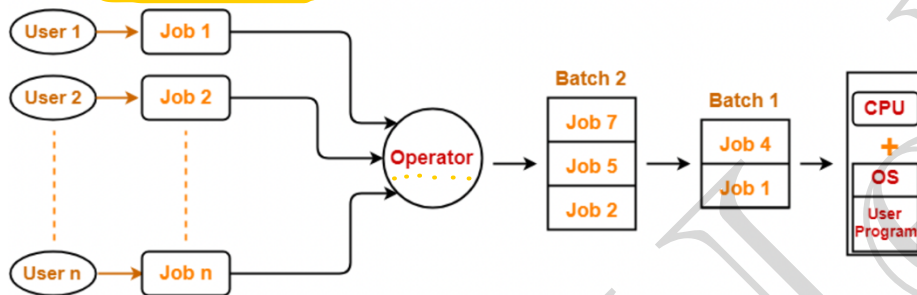
Multiprogramming increases CPU utilization by keeping multiple jobs (code and data) in the memory so that the CPU always has one to execute in case some job gets busy with I/O.

PCB - Process Control Block stores process's current state

- Single CPU
- Context switching for processes.
- Switch happens when current process goes to wait state.
- CPU idle time reduced.

ready queue

**Multitasking** is a logical extension of multiprogramming.
- Single CPU
- Able to run more than one task simultaneously.   + time sharing.
- Context switching and time sharing used.
- Increases responsiveness.
- CPU idle time is further reduced.
    - process starvation is less.

time quantum ex: 100ms

Multi-processing OS, more than 1 CPU in a single computer.

context switching + time sharing + ...

- Increases reliability, 1 CPU fails, other can work
- Better throughput.
- Lesser process starvation, (if 1 CPU is working on some process, other can be executed on other CPU.
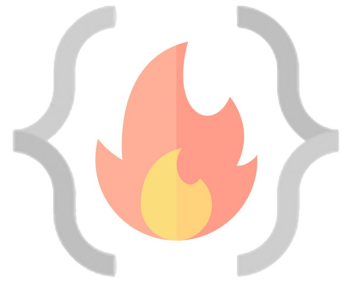
Distributed OS,
- OS manages many bunches of resources, >=1 CPUs, >=1 memory, >=1 GPUs, etc
- **Loosely connected autonomous,** interconnected computer nodes.
- collection of independent, networked, communicating, and physically separate computational nodes.

RTOS
- **Real time** error free, computations within tight-time boundaries.
- Air Traffic control system, ROBOTS etc. (self driving)