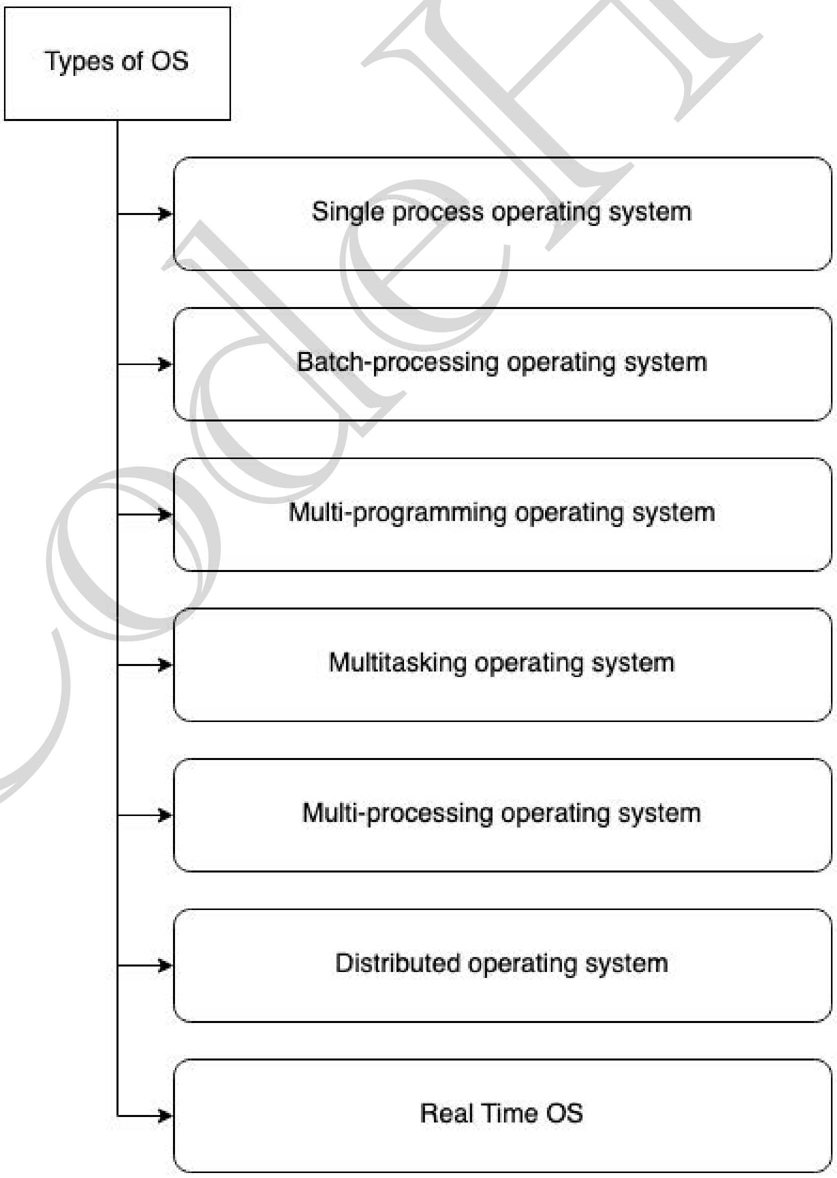
LEC-2: Types of OS

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]

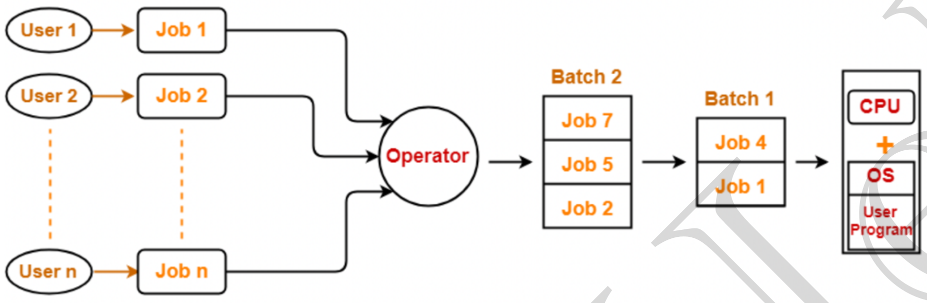

Types of OS
Single process operating system
Batch-processing operating system
Multi-programming operating system
Multitasking operating system
Multi-processing operating system
Distributed operating system
Real Time OS

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.

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

PCB(process control block): A data structure that saves the state of process while context switching.

Multitasking is a logical extension of multiprogramming.

- Single CPU
- Able to run more than one task simultaneously.
- Context switching and time sharing used.
- Increases responsiveness.
- CPU idle time is further reduced.

time limit to every job instead of waiting for a job to go over to input/output state so that another job can be taken by CPU.

high priority job is addressed through context switching.

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

- 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.

context sharing, time sharing, CPU>=1

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.

os connected to multiple cpu in a network.
can have multiple sources.

RTOS

- **Real time** error free, computations within tight-time boundaries.
- Air Traffic control system, ROBOTS etc.