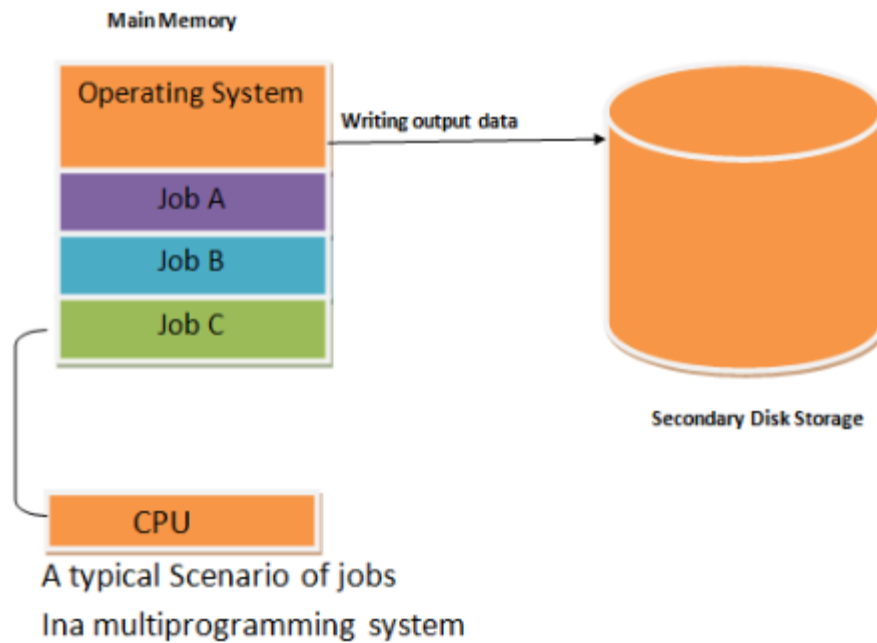# Multiprogramming Operating System

- In a multiprogramming system there are one or more programs loaded in main memory which are ready to execute. Only one program at a time is able to get the CPU for executing its instructions (i.e., there is at most one process running on the system) while all the others are waiting their turn.

- The main idea of multiprogramming is to maximize the use of CPU time.

- Indeed, suppose the currently running process is performing an I/O task (which, by definition, does not need the CPU to be accomplished).

- Then, the OS may interrupt that process and give the control to one of the other in-main-memory programs that are ready to execute (i.e. *process context switching*).

- In this way, no CPU time is wasted by the system waiting for the I/O task to be completed, and a running process keeps executing until either it voluntarily releases the CPU or when it blocks for an I/O operation.

- Therefore, the ultimate goal of multiprogramming is to keep the CPU busy as long as there are processes ready to execute.

- Note that in order for such a system to function properly, the OS must be able to load multiple programs into separate areas of the main memory and

provide the required protection to avoid the chance of one process being modified by another one.

Main Memory

Operating System

Writing output data →

Job A
Job B
Job C

CPU

Secondary Disk Storage

A typical Scenario of jobs
Ina multiprogramming system

- Other problems that need to be addressed when having multiple programs in memory is *fragmentation* as programs enter or leave the main memory.

- Another issue that needs to be handled as well is that large programs may not fit at once in memory which can be solved by using *pagination* and *virtual memory*. Please, refer to this article for more details on that.

- Finally, note that if there are `N` ready processes and all of those are highly CPU-bound (i.e., they mostly execute CPU tasks and none or very few I/O operations), in the very worst case one program might wait all the other `N-1` ones to complete before executing.

/

OPERATING SYSTEM   TYPES OF OPERATING SYSTEM

www.tutorialsspace.com
B·Tech/B·E- BCA-MCA –CSE-IT
GATE- UGC-NET– PSU UNIV·EX

MULTIPROGRAMMING OS

In this OS Keeps Several jobs/Programs in memory So CPU Select a job and Start Executing it, when that job needs to wait for any i/o operations then CPU Switch to another job & Start executing it.
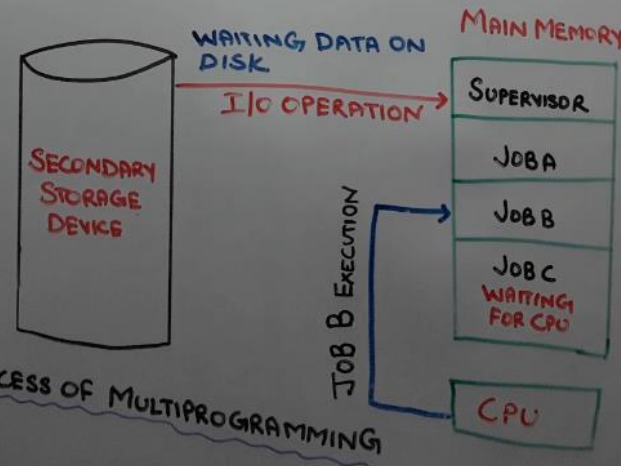So By this CPU keeps Busy & 'highly Utilized' & never sit 'IDLE'

ADVANTAGES:-
→ HIGH CPU UTILIZATION
→ INCREASED THROUGHPUT
→ SHORTER RESPONSE TIME
→ SHORTER WAITING TIME
→ ABILITY TO ASSIGN PRIORITY TO JOBS.
→ MANY SCHEDULING ALGORITHMS CAN BE USED.

DISADVANTAGES
→ SCHEDULING IMPLEMENTATION IS NOT EASY
→ MORE MANAGEMENT IS REQUIRED
    → MEMORY M|G
    → CONTEXT SWITCHING
    → SCHEDULING

WAITING DATA ON DISK

I/O OPERATION

SECONDARY STORAGE DEVICE

MAIN MEMORY

SUPERVISOR
JOB A
JOB B
JOB C WAITING FOR CPU

JOB B EXECUTION

CPU

PROCESS OF MULTIPROGRAMMING