

Department of Computer Science and Engineering

**FACULTY OF ENGINEERING AND TECHNOLOGY
UNIVERSITY OF LUCKNOW
LUCKNOW**



Dr. Zeeshan Ali Siddiqui
Assistant Professor
Deptt. of C.S.E.

MULTIPLE-PROCESSOR SCHEDULING CONCEPT

Multiple-Processor Scheduling

- CPU scheduling more complex when **multiple** CPUs are available
 - **Homogeneous** processors within a multiprocessor
- **Asymmetric multiprocessing** – Only one processor accesses the system data structures, reducing the need for data sharing.
- **Symmetric multiprocessing** – Each processor is self-scheduling, all processes in common ready queue, or each has its own private queue of ready processes.

Processor Affinity

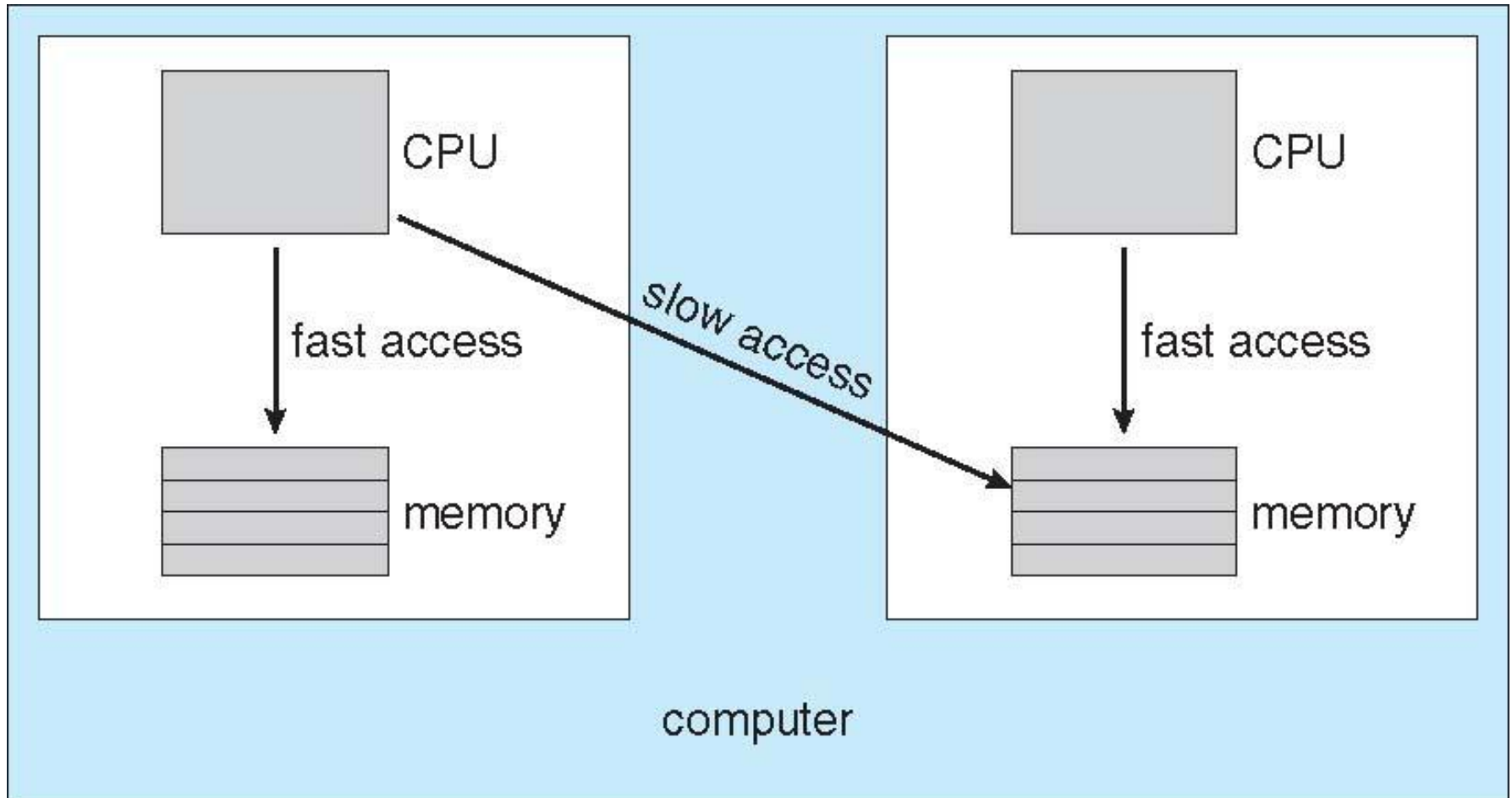
- Process has **affinity** for processor on which it is currently running (e.g., to avoid repopulating caches)
- **Soft affinity**- When an operating system has a policy of attempting to keep a process running on the same processor—but not **guaranteeing** that it will do so.
- **Hard affinity**- Allowing a process to specify a **subset** of processors on which it may run.

Load Balancing

- **Push migration** - A specific task periodically checks the load on each processor and, if it finds an **imbalance**, evenly distributes the load by moving (or pushing) processes from overloaded to idle or less-busy processors.
- **Pull migration** - Pull migration occurs when an **idle** processor pulls a waiting task from a busy processor.

NUMA and CPU Scheduling^{1/2}

- NUMA - *Non-uniform memory access*
- CPU Scheduling - Architecture can affect processor affinity



NUMA and CPU Scheduling^{2/2}

- NUMA, in which a CPU has faster **access** to some parts of main memory than to other parts.
 - Typically, this occurs in systems containing **combined** CPU and memory boards.
 - The CPUs on a board can access the memory on that board **faster** than they can access memory on other boards in the system.
- If the operating system's CPU scheduler and memory-placement algorithms work together, then a process that is having **affinity** to a particular CPU can be allocated to the same CPU.

References

1. Silberschatz, Galvin and Gagne, “Operating Systems Concepts”, Wiley.
2. William Stallings, “Operating Systems: Internals and Design Principles”, 6th Edition, Pearson Education.
3. D M Dhamdhere, “Operating Systems: A Concept based Approach”, 2nd Edition, TMH.

Thank You.

