



Module 6

Scheduling

▼ What are scheduling policies?

algorithms for allocating CPU resources to concurrent tasks deployed on a processor (i.e., computing resource) or a shared pool of processors

▼ Why use scheduling?

to allocate available time and processors to a job or a process

▼ Why are uniprocessor systems easier to use compared to multiprocessor systems?

For a uniprocessor system, there is ease of programming which is easy to support kernel synchronization and CPU scheduling.

▼ Why do multiprocessor systems perform better than uniprocessor systems?

since on a single computing unit, all operations (additions, multiplications, other arithmetic, logical operations, etc.) are done sequentially on the unit while they can be joined together in a multiprocessor environment

▼ Example scheduling policies?

- Earliest deadline first EDF scheduling
- First in first out (FIFO) scheduling
- Fixed priority schedule
- Lead laxity first scheduling
- Round robin scheduling
- Timetable driven scheduling

▼ Static vs. Dynamic Scheduling

- A static scheduler allocates how many processors to use one-time. The static method provides low run-time scheduling and reduces overhead.
- Dynamic scheduling allows for altering the number of threads used in a process throughout the course of its execution. This may cause higher run-time scheduling and thus create more overhead resource requirements, but it can also free up and allow for redistribution of unused resources.

▼ Components of multiprocessor systems

- The loosely coupled system consists of a collection of relatively autonomous systems connected with an interconnection network. Each has its own memory and I/O Channels.
- With the functionally specialized processors there is a master, general-purpose processor. Specialized processors are controlled by the master processor and provide services to it.
- Tightly coupled multiprocessors consist of a set of processors that share a common main memory under the integrated control of an operating system.

▼ What is load sharing?

- processors are not assigned to a particular processor
- it carries over directly from a uniprocessor environment

▼ What is gang scheduling?

- A set of related threads scheduled to run on a set of processors at the same time on a one-to-one basis.
- It is useful for medium-grained to fine-grained parallel applications whose performance degrades when any part of the application is not running while other parts are ready to run.

▼ What does dedicated processor assignment provide?

implicit scheduling defined by the thread assignment to processes through the entire course of execution

▼ What is dynamic scheduling?

It allows for a number of threads of a given process to be altered during the course of execution

▼ What are real-time processes?

those in which the correctness of the system depends on the logical result of the computation and also the time at which the results are produced

▼ What are the requirements for real-time operating systems?

- **Determinism:** Deals with the time taken for an operating system to acknowledge an interruption. The operations here are done at times that have been predetermined; however, when there are multiple processes competing for resources, the system cannot be fully deterministic.
- **Responsiveness:** How long it takes the operating system to service the interrupt.
- **User control:** A user should be able to distinguish between hard and soft tasks and to specify relative priorities within each class.
- **Reliability:** Loss or degradation of performance may have catastrophic consequences such as financial loss or loss of life.
- **Fail-soft operation:** The ability of a system to fail in such a way that it preserves as much capability and data as possible.

▼ What are the different times in a scheduling process?

- arrival time, e.g. the time you get in line
- burst time, e.g. the time it takes to receive the order
- completion time, e.g. the time it takes to leave a restaurant
- turnaround time, or completion - arrival time

▼ What is the first come first served method?

The processor picks up the ready thread. Threads are arranged consecutively in a shared queue.

▼ What is the smallest number of threads first method?

The priority is given to threads from jobs with the smallest number of unscheduled threads. When they have the same number of threads, it follows the first come first serve schedule.

▼ What is the preemptive smallest number of threads first method?

Priority is given to jobs with the smallest number of unscheduled threads. An arriving job with a smaller number of threads than an executing job will preempt threads belonging to the scheduled job.