Process Concept & Scheduling

Nirmala Shinde Baloorkar
Assistant Professor
Department of Computer Engineering

SOMAIYA
VIDYAVIHAR UNIVERSITY
K J Somaiya College of Engineering

Somaiya TRUST

Outline

- Basic Concept
- Scheduling Criteria
- Scheduling Algorithm

Process Scheduling

- The operating system is responsible for managing the *scheduling* activities.
  - A uniprocessor system can have only one running process at a time
  - The main memory cannot always accommodate all processes at run-time
  - The operating system will need to decide on which process to execute next (CPU scheduling), and which processes will be brought to the main memory (job scheduling)

Process Scheduling

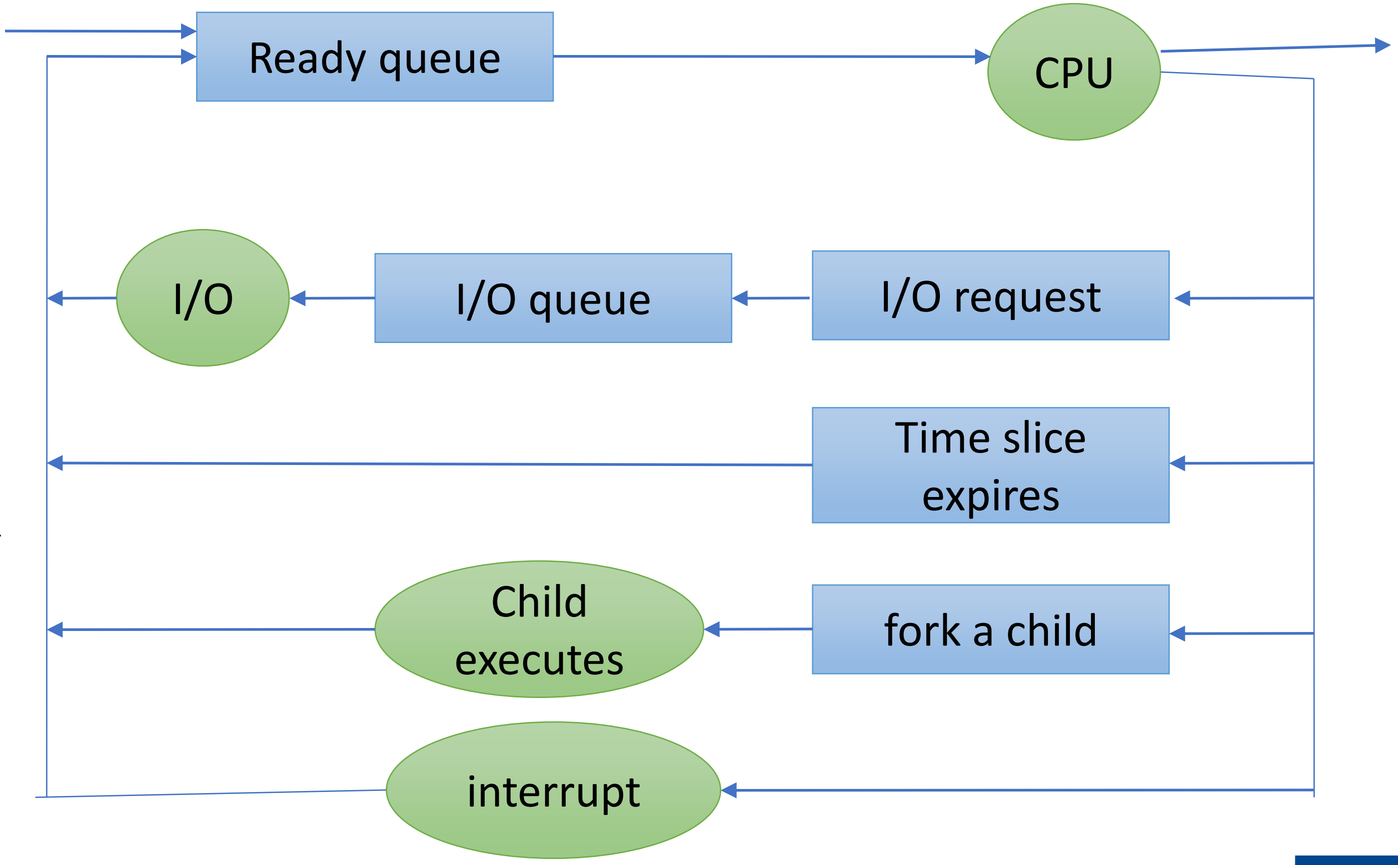
Process Scheduler – selects one of the available processes for execution

Queues

Job queue – all processes in system	Ready queue – all ready processes	Device queue – process waiting for a device

Process Lifecycle

- There are Two queues ready queue and device queue.
- A new process is first admitted into the ready queue.
- It waits until it is **dispatched** (allocated to the CPU for execution).



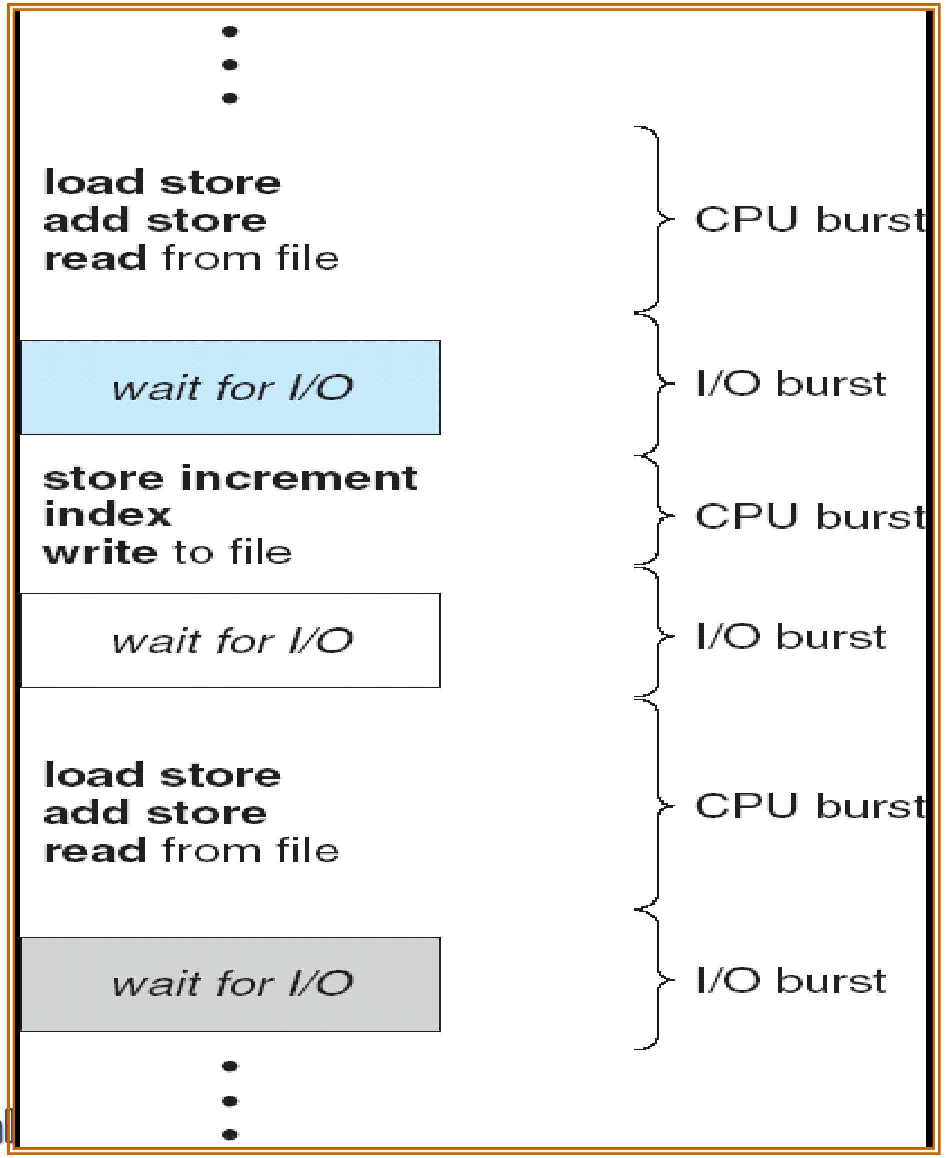
Process Type

CPU bound

I/O bound

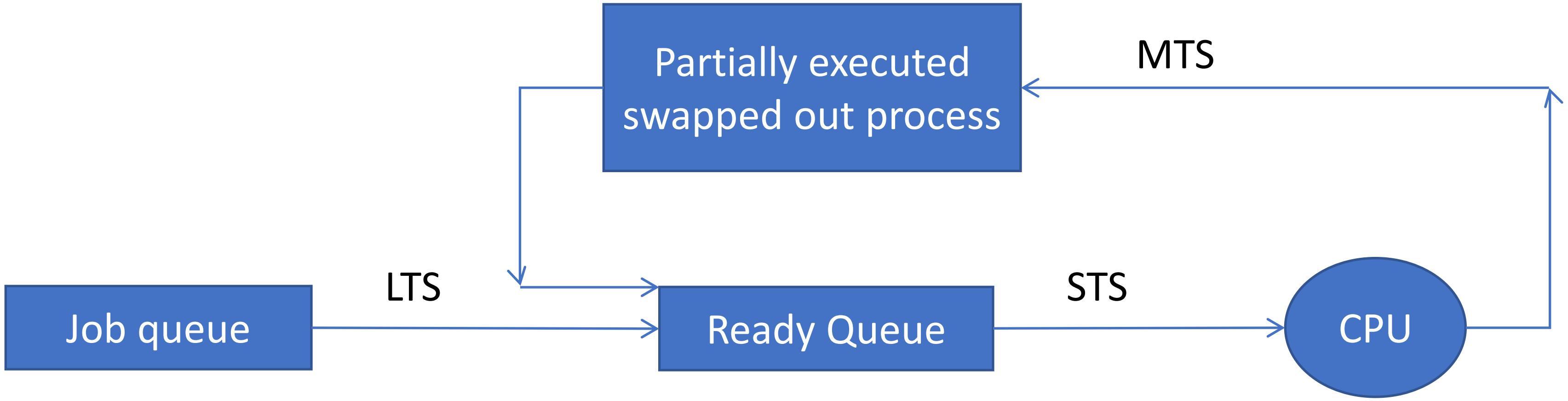
CPU and I/O Bursts

- CPU–I/O Burst Cycle –
  - Process execution consists of a *cycle* of CPU execution and I/O wait.
- I/O-*bound process* – spends more time doing I/O than computations, many short CPU bursts.
- *CPU-bound process* – spends more time doing computations; few very long CPU bursts.



Schedulers

- Short Term Scheduler (STS)
- Long Term Scheduler (LTS)
- Medium Term Scheduler (MTS)

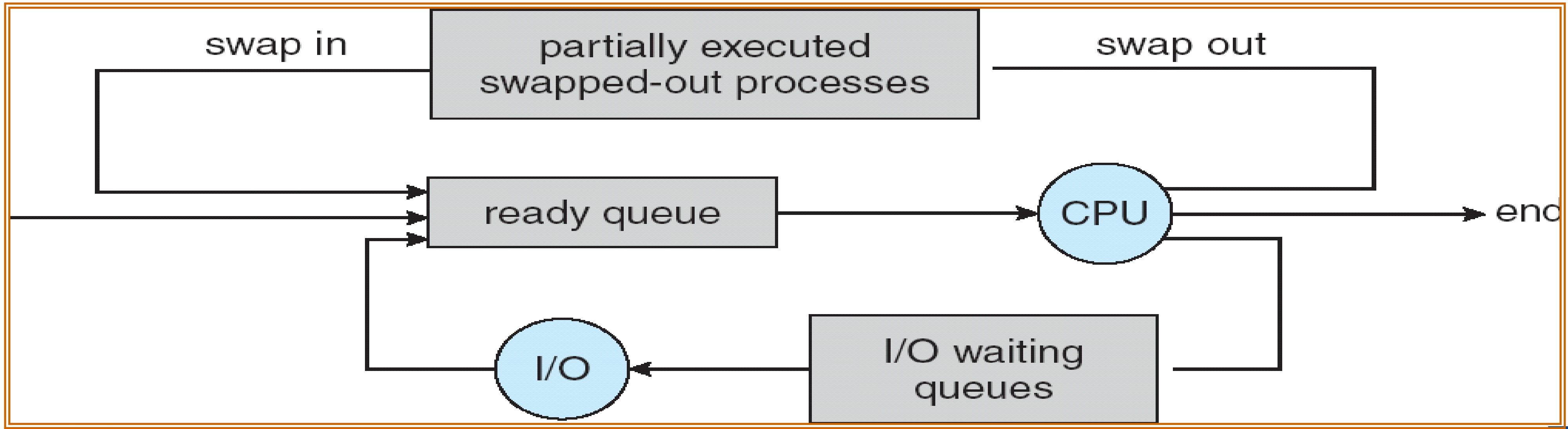


Schedulers (continued)

- The processes may be first spooled to a mass-storage system, where they are kept for later execution.
- The *long-term scheduler (or job scheduler)* – selects processes from this pool and loads them into memory for execution.
  - The long term scheduler, if it exists, will control the *degree of multiprogramming*
- The *short-term scheduler (or CPU scheduler)* – selects from among the *ready* processes, and allocates the CPU to one of them.
  - Unlike the long-term scheduler, the short-term scheduler is invoked very frequently.

Addition of Medium-Term Scheduler

- Controlling the degree of multiprogramming by swapping processes in and out of the main memory.
- This helps manage the number of processes in the system at any given time, maintaining an optimal level of multiprogramming and balancing the system's load.



Scheduler Impact

- Long-term (job) scheduler decisions – want a mix of CPU- and IO-bound processes
- Short-term (CPU) scheduler decisions
- Consequences of using I/O-bound and CPU-bound process information

Comparison of Schedulers

Parameters	Long-Term	Short-Term	Medium-Term
Type of Scheduler	It is a type of job scheduler.	It is a type of CPU scheduler.	It is a type of process swapping scheduler.
Speed	Its speed is comparatively less than that of the Short-Term scheduler.	It is the fastest among the other two.	Its speed is in between both Long and Short-Term schedulers.
Minimal time-sharing system	Almost absent	Minimal	Present

Comparison of Schedulers

Parameters	Long-Term	Short-Term	Medium-Term
Purpose	A Long-Term Scheduler helps in controlling the overall degree of multiprogramming.	The Short-Term Scheduler provides much less control over the degree of multiprogramming.	Medium-Term reduces the overall degree of multiprogramming.
Function	Selects processes from the pool and then loads them into the memory for execution.	Selects all those processes that are ready to be executed.	Can re-introduce the given process into memory. The execution can then be continued.

CPU-scheduling decision circumstances

- CPU scheduling means to select the next process to be allocated to the CPU whenever the CPU becomes idle.
- Other common names of CPU scheduling are **process scheduling or thread scheduling**.

1. A Process switches from running state to waiting state.
2. Process switches from running state to ready state.
3. Process switches from waiting state to ready state.
4. A Process terminates.

1 and 4 are **Non-Preemptive.**
2 and 3 are **Preemptive**.

Comparison

Non-Preemptive scheduling	Preemptive Scheduling
• Once the CPU has been allocated to a process, the process keeps the CPU until it releases the CPU either by terminating or by switching to the waiting state. • Voluntarily releases the CPU	• The CPU is allocated to the new process if it has higher priority as compared to current process or if the time slice expires. • Forced to release the CPU

Dispatcher

- Gives control of the CPU to the process.
- It involves
  - Context switching
  - Switching to user mode
  - Jumping to the proper location in the user program to restart that process.

Scheduling Criteria

Several criteria can be used to compare the performance of scheduling algorithms

- CPU utilization – keep the CPU as busy as possible
- Throughput – # of processes that complete their execution per time unit
- Turnaround time – amount of time to execute a particular process
- Waiting time – amount of time a process has been waiting in the ready queue
- Response time – amount of time it takes from when a request was submitted until the first response is produced, not the complete output.
- Fairness - Ensuring that all processes receive an equitable share of CPU time, preventing any single process from monopolizing resources or starving.
- Meeting the deadlines (real-time systems)

Optimization Criteria

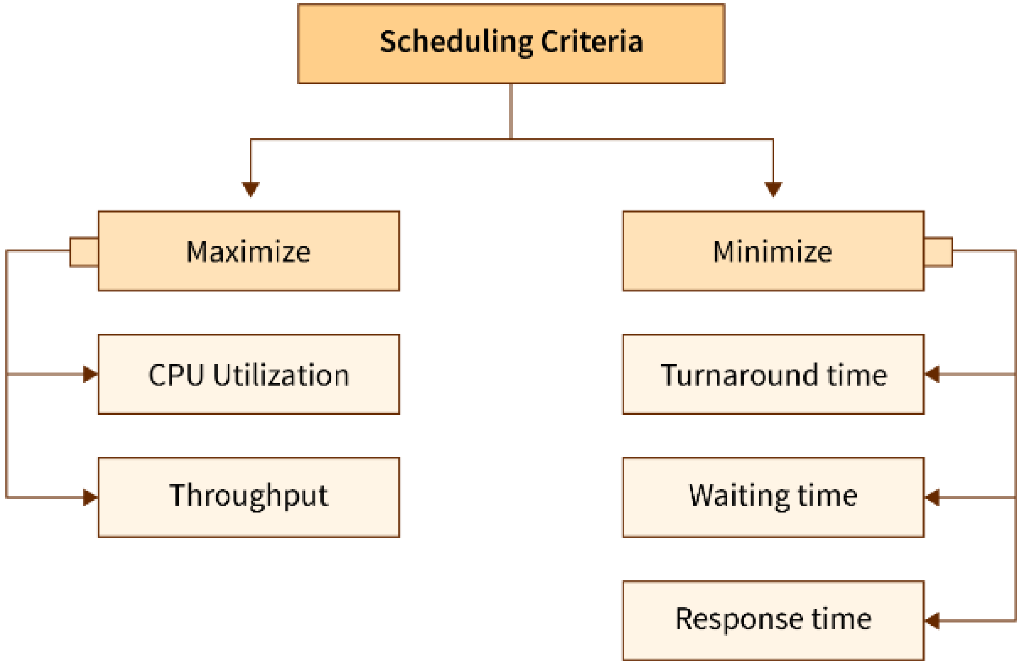


Image source : Scaler

- In the examples, we will assume
  - average waiting time is the performance measure
  - only one CPU burst (in milliseconds) per process

Question ?