Operating System : Scheduling

Prepared By

Dr. Sanjeev Gangwar

Assistant Professor

Department of Computer Applications

VBS Purvanchal University, Jaunpur, Uttar Pradesh

Scheduling

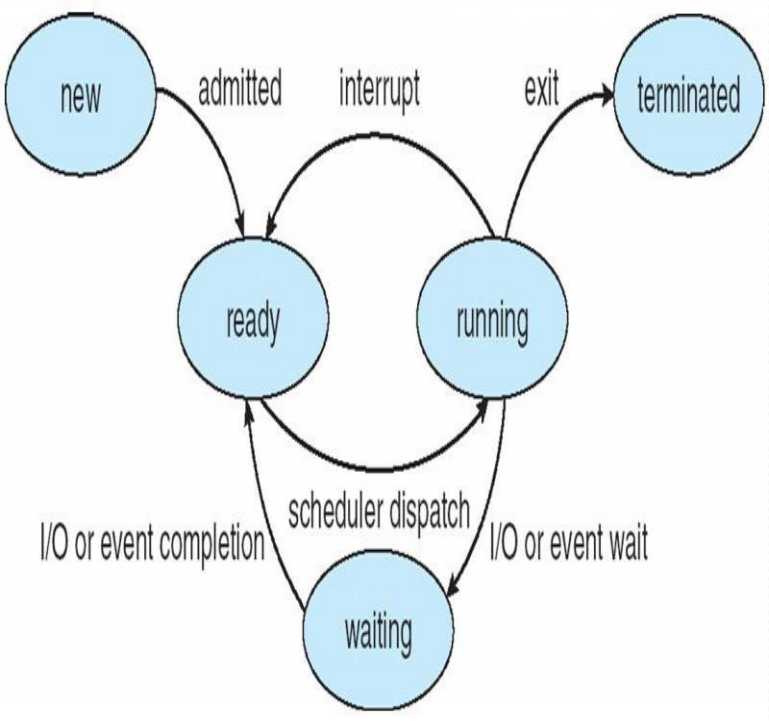
- Basic Concepts
- Recall
- Process Control Block
- CPU and IO Bursts
- Process Scheduling
- CPU Scheduler
- Dispatcher
- Scheduling Criteria
- Optimization Criteria
- Types of Schedulers
- Comparison between Scheduler
- References

Basic Concepts

- Maximum CPU utilization obtained with multiprogramming.
- Scheduling refers to a set of policies and mechanisms built into the operating system that govern the order in which the work to be done by a computer system.
- A scheduler is an operating system module that selects the next jobs to be admitted into the system and the next process to run.
- CPU-I/O Burst Cycle- Process execution consists of a cycle of CPU execution and I/O wait.

Recall

A process is an instance of a program in execution. As a process executes, it changes state.



Process States

- **New:** the process is being created.
- **Ready:** the process is waiting to be assigned to a processor.
- **Waiting:** the process is waiting for some event to occur.
- **Running:** instructions are being executed
- **Terminated:** the process has finished execution

Process Control Block

Each Process is represented in the operating system by a process control block(PCB) also called a task control block. It contains many pieces of information associated with a specific process including these

- Process state
- Program Counter
- CPU registers
- CPU scheduling information
- Memory management information
- Accounting information
- I/O status information

CPU and IO Bursts

.
.
.
load, store,
add, store,
read from file

Wait for IO

Store, increment,
branch, write to file

Wait for IO

load, store,
read from file

Wait for IO

.
.
.

- **CPU–I/O Burst Cycle** – Process execution consists of a cycle of CPU execution and I/O wait
- Process execution consists of a cycle of CPU execution and I/O wait. Processes alternate between these two states. Process execution begins with a CPU burst. That is followed by an I/O burst, which is followed by another CPU burst, then another I/O burst, and so on.
- The last CPU burst will end with a system request to terminate execution, rather than with another I/O burst.

Process Scheduling

CPU is always busy in Multiprogramming. Because CPU switches from one job to another job. scheduling is a important OS function. All resources are scheduled before use.

Scheduling Queues

Job Queue: when processes enter the system, they are put into a job queue, which consists all processes in the system. Processes in the job queue reside on mass storage and await the allocation of main memory.

Ready Queue: if a process is present in main memory and is ready to be allocated to cpu for execution, is kept in ready queue.

Device Queue: if a process is present in waiting state (or) waiting for an i/o event to complete is said to be in device queue.

CPU Scheduler

Selects from among the processes in memory that are ready to execute, and allocates the CPU to one of them. CPU scheduling decisions may take place when a process:

- From running state to waiting state
- From running state to ready state
- From waiting state to ready state
- Terminates

Scheduling under 1 and 4 is nonpreemptive All other scheduling is preemptive

Dispatcher

- **Dispatcher:** The main job of dispatcher is switching the cpu from one process to another process. Dispatcher connects the cpu to the process selected by the short term scheduler.
- **Dispatcher latency:** The time it takes by the dispatcher to stop one process and start another process is known as dispatcher latency. If the dispatcher latency is increasing, then the degree of multiprogramming decreases.

Scheduling Criteria

- **CPU utilization:** CPU is costly device, it must be kept as busy as possible. Eg: CPU efficiency is 90% means it is busy for 90 units, 10 units idle.
- **Throughput:** Number of process that are completed per time unit
- **Arrival Time(AT)**: Time at which the process arrives in the ready queue.
- **Turnaround time(TAT)**: Time between submission and completion
- **Waiting time(WT):** The time spent by the process to wait for CPU to be allocated.
- **Response time:** Time between submission and first response.
- **Completion Time(CT):** The Time at which the process enters into the **completion** state or the time at which the process completes its execution.

Optimization Criteria

- Max CPU utilization
- Max throughput
- Min turnaround time
- Min waiting time
- Min response time

Types of Schedulers

There are three types of schedulers

- **Long term scheduler:** select the jobs from the job pool and loaded these jobs into main memory (ready queue). Long term scheduler is also called job scheduler.
- **Short term scheduler:** select the process from ready queue, and allocates it to the cpu. short term scheduler maintains ready queue, device queue. Also called as cpu scheduler.
- **Medium term scheduler:** if process request an I/O device in the middle of the execution, then the process removed from the main memory and loaded into the waiting queue. When the I/O operation completed, then the job moved from waiting queue to ready queue. These two operations performed by medium term scheduler.

Comparison Between Scheduler

S.No.	Long Term Scheduler	Short Term Scheduler	Medium Term Scheduler
1	It is a job scheduler	It is a CPU scheduler	It is a process swapping scheduler.
2	Speed is lesser than short term scheduler	Speed is fastest among two	Speed is in between both short and long term scheduler.
3	It controls the degree of multiprogramming	It provides lesser control over degree of multiprogramming	It reduces the degree of multiprogramming.
4	It is almost absent or minimal in time sharing system	It is also minimal in time sharing system	It is a part of Time sharing systems.
5	It select processes from pool and loads them into memory for execution	It selects those processes which are ready to execute	It can re-introduce the process into memory and execution can be continued.

References

- Abraham Silberschatz, Galvin & Gagne, Operating System Concepts, John Wiley & Sons, INC.
- Harvay M.Deital, Introduction to Operating System, Addition Wesley Publication Company.
- Andrew S.Tanenbaum, Operating System Design and Implementation, PHI
- Vijay Shukla, Operating System, S.K. Kataria & Sons

Thank You