SCHEDULING
OR
PROCESSOR SCHEDULING

In a multiprogramming Operating System, the method or procedure for switching the CPU among multiple processes is called CPU scheduling or processor scheduling.

The CPU schedular is a part of Operating System which is responsible for CPU scheduling. Whenever CPU becomes idle, the CPU schedular selects a process from the ready queue and send it to the CPU for execution.

For Example- Process A is in running state and need to perform I/O related task. process A doesnot need the CPU while performing the I/O task.

PROCESSOR SCHEDULAR

Processor schedular change the state of process A from running to waiting state and enables process B to use the CPU.

For Example- In the process of CPU scheduling, two processes A and B share the CPU. When the process A uses the CPU, the process B perform I/O operation and when the CPU is accessed by process B, the process A perform I/O operation.

Scanned by CamScanner

DULAR

Schedular is a part of Operating System which is responsible for entire process scheduling and transition of process state.

LEVELS/TYPES OF SCHEDULAR

<1> Long term schedular
<2> Medium term schedular
<3> Short term schedular.

<1> Long-term schedular ⇒ Long term schedular works with a new process which should be initiated in the system. Once initiated, this schedular takes the newly submitted jobs and convert them into processes, which are further put in the ready queue for processing.

<2> Medium-term schedular ⇒ Medium term schedular decides whether to introduce a process to the waiting state from the running state or put up back a process from waiting state to the ready state.

<3> Short-term schedular ⇒ Schedular decision are made on which process in the ready state should be dispatched to the CPU/running state for execution.

This schedular is the most important and complex among all the scheduling levels.

Scanned by CamScanner

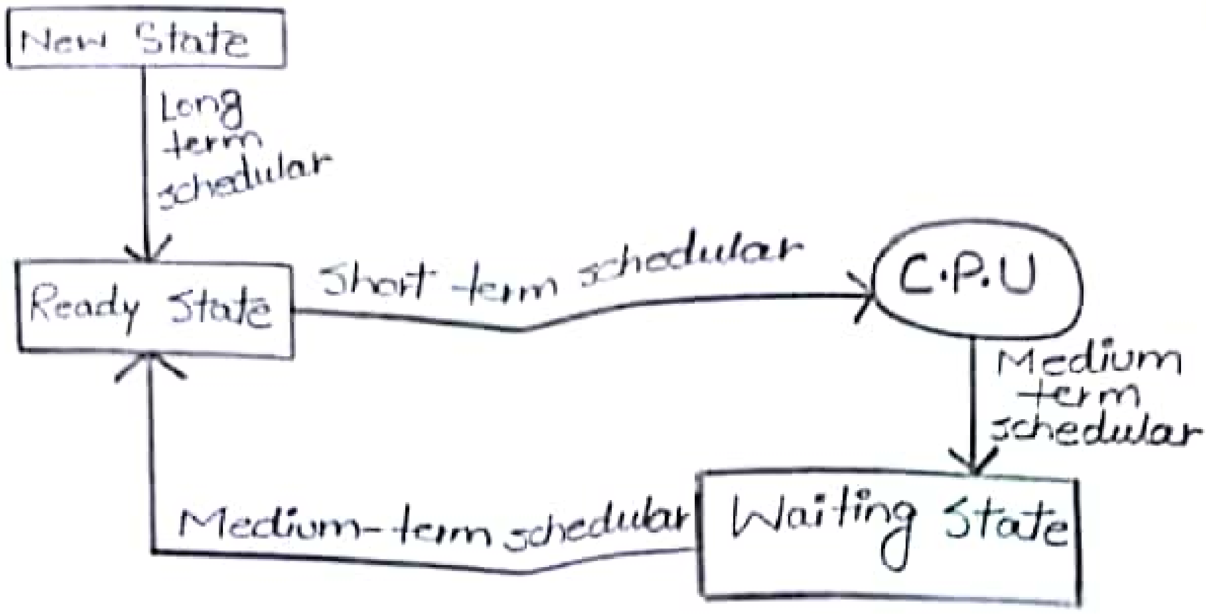


Fig. Schedular

SCHEDULING CRITERIA

Some performance criteria that are frequently used by schedular to maximize system performance are as follows:-

1) CPU Utilization ⇒ The key idea is that if the CPU is busy all the time, the utilization factor of all the components of the system will also be high.

2) Throughput ⇒ It refers to the amount of work completed in a unit of time.

One way to measure throughput is by means of the number of processes that are completed in a unit of time. The higher the number of processes, the more work apparently being done by the system.

Scanned by CamScanner

naround Time ⇒ The amount of time to complish the execution of a process is known as turnaround time of the process. It is always minimized.

Turnaround time = Processing time/Execution time + Waiting time.

4.) Waiting Time ⇒ The amount of time a process waits in the ready queue to get a chance for execution is known as waiting time of the process.

Waiting time = Turnaround time - Processing/Execution time

5.) Response Time ⇒ It is the amount of time, it takes to start responding, not the time taken to output that response.

SCHEDULING ALGORITHMS

1. CPU scheduling deals with the problem of deciding which of the processes in the ready queue to be allocated the CPU.

2. There are several scheduling algorithm for taking decision.

3. A major division among scheduling algorithm is that whether they support preemptive or non-preemptive scheduling:-

Scanned by CamScanner

nonpreemptive scheduling ⇒

i) A scheduling discipline is nonpreemptive, if once a process has given to C.P.U, the C.P.U cannot be taken away from that process.

ii) Ex → DOS, Window 3.1 etc.

iii) The control of the C.P.U is in the hand of process

iv) In nonpreemptive, system jobs are made to wait by longer time, but the treatment of the process is fairer.

(B) Preemptive scheduling ⇒

i) A scheduling discipline is preemptive if the C.P.U can be taken away.

ii) The control of the C.P.U is in the hand of Operating system.

iii) Ex → Win' 95 upto is preemptive.

iv) Preemptive scheduling is useful in high priority process which require immediate response.

Several types of CPU scheduling algorithm are:-

(1) FCFS (First-Come, First-Served) Scheduling Algorithm.

(2) SJF (Shortest - Job - First) Scheduling Algorithm.

(3) SRTN (Shortest Remaining Time Next) Scheduling Algorithm.

(4) Priority scheduling

(5) RR (Round-Robin) Scheduling Algorithm

Scanned by CamScanner

FCFS (First-Come, First-Served) Scheduling Algorithm

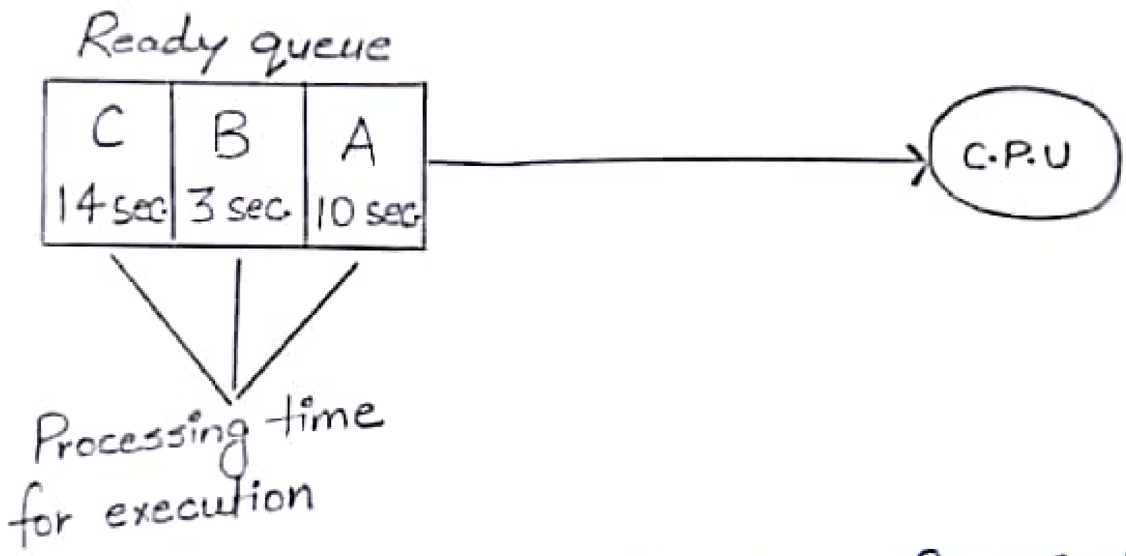
(i) FCFS algorithm is the simplest and most state forward of all the CPU scheduling algorithm.

(ii) A process is allocated CPU time according to the arrival time of a process. The process that comes first is allocated the CPU first.

(iii) A FCFS scheduling is non-preemptive which usually results in poor performance.

(iv) This results in poor scheduling performance because a process requiring very short time waits for the completion of a process that requires a comparatively longer time.

Example-



Turnaround time of process A = Waiting time + Processing time
= 0 + 10
= 10 sec.

Turnaround time of process B = Waiting time + Processing time
= 10 + 3
= 13 sec.

Scanned by CamScanner

...und time of process C = Waiting time + Processing time
= 13 + 14
= 27 sec.

In FCFS scheduling, various processes are scheduled for execution regardless of their accepted execution time.

The process A comes first in the ready queue followed by the processes B and C.

The process are executed in a sequence. First, A is executed, then B and then C.

<2> SJF (Shortest - Job - First) Scheduling Algorithm

(i) It is faster than FCFS.

(ii) In SJF, the process with the least estimated execution time is selected from the ready queue for execution.

(iii) This algorithm is non-preemptive in nature.

(iv) It check always for shorted executed time.

(v) In SJF scheduling, various processes are scheduled for execution according to their estimated execution time.

Example

Turnaround time of process B = Waiting time + Processing time
= 0 + 3
= 3 sec.

Scanned by CamScanner

ound time of process A = Waiting time + Processing time

= 3 + 10

= 13 sec.

Turnaround time of process C = Waiting time + Processing time

= 13 + 14

= 27 sec.

The process A, B, C requires 10, 3 and 14 sec. respectively.

The process B with minimum estimated execution time is scheduled for execution first followed by A and C.

<3> SRTN (Shortest Remaining Time Next) Scheduling Algorithm

(i) It is a scheduling discipline in which the next scheduling entity, a job or processes is on the basis of the shortest remaining execution time.

(ii) SRTN scheduling may be implemented either in the non-preemptive or preemptive variety.

(iii) The Non-preemptive of SRTN is called SJF (Shortest - Job - First) scheduling.

(iv) This scheduling searches the shorted remaining execution time. The difference between two cases lies in the condition that lead in evacuating the schedular.

Scanned by CamScanner

out preemption, SRTN schedular control to the process.

In the preemption version, whenever an event occurs, that makes a new process to the ready queue and control directly to the operating system

(vii) SRTN is probably an optimal scheduling discipline in terms of minimizing the average waiting time of a given work load.

(viii) It is the preemption version of SJF (Shortest - Job - First) Scheduling with fixed interval time for each process.

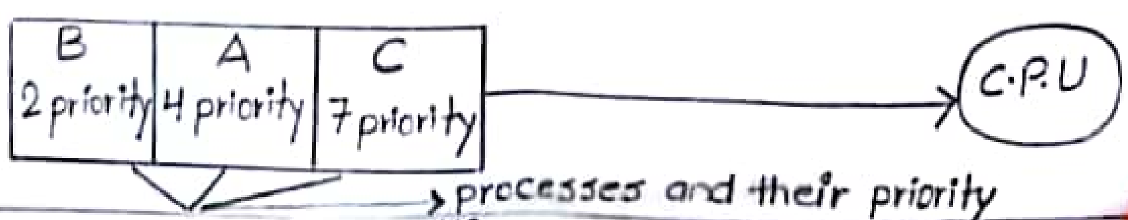
(4) Priority Scheduling

(i) In priority scheduling, a priority is associated with all processes. Processes are executed in a sequence according to their priority.

(ii) The CPU time is allocated to the process with the highest piority.

(iii) If the priority of two or more process are equal then the process that has been inserted first into the ready queue is selected for execution.

In other word, FCFS scheduling is performed.



Scanned by CamScanner

priority scheduling algorithm, various processes are scheduled for execution according to their priority such as A = 4 priority
B = 2 priority
C = 7 priority

(v) The process C with the highest priority is selected for execution first followed by process A and B.

<5> RR (Round-Robin) Scheduling Algorithm

(i) The RR scheduling algorithm implement the time sharing concept.

(ii) Each process is allow to use C.P.U for a predefine unit of time known as time-slice or interval time with the help of a interval time hardware.

(iii) If the running process is completed within the time slice, the next process in the ready queue is allocated to the C.P.U.

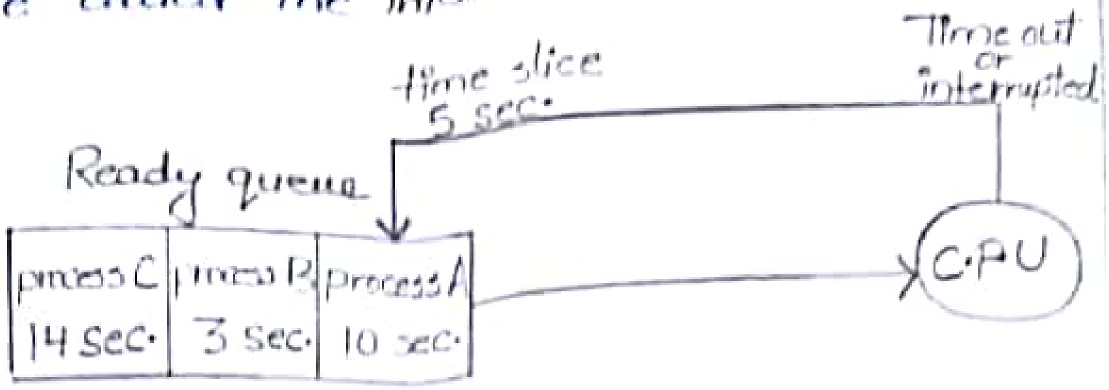
(iv) In case, if the running process remain incomplete, it is stored back to the ready queue and the next process in the ready queue is selected for the execution.

(v) In Round-Robin scheduling algorithm, the ready queue is a FCFS queue. When the time out occurs, the partly done processes are inserted at the end of the ready queue.

Scanned by CamScanner

Is similar to FCFS but preemptive in nature. A timer is used to mark the end of a time slice called the interval timer.

Example-



Processing
⇓

process C 4 sec	process C 5 sec	process A 5 sec.	process C 5 sec	process B 3 sec	process A 5 sec

Waiting time of process A = 8 sec

Processing time of process A = 10 sec

Turnaround time of process A = Waiting time + Processing time

= 8 + 10

= 18 sec.

Waiting time of process B = 5 sec.

Processing time of process B = 3 sec.

Turnaround time of process B = Waiting time + Processing time

= 5 + 3

= 8 sec.

Scanned by CamScanner

g time of process C = 13 sec.

essing time of process C = 14 sec.

around time of process C = Waiting time + Processing time

= 13 + 14

= 27 sec.

In figure, there are three process A, B and C. Each process have a time slice 5 sec. The operating system transfer the control of the CPU to the next process after 5 sec. automatically.

INTERVAL TIMER

Timer interruption is a technique that is closely related to preemption. When a process gets the CPU, a timer may be set to a specified interval. If the process is still using the CPU at the end of the interval, then it is preempted. Both timer interruption and preemption force a process to yield the CPU before its CPU burst is complete.

Scanned by CamScanner