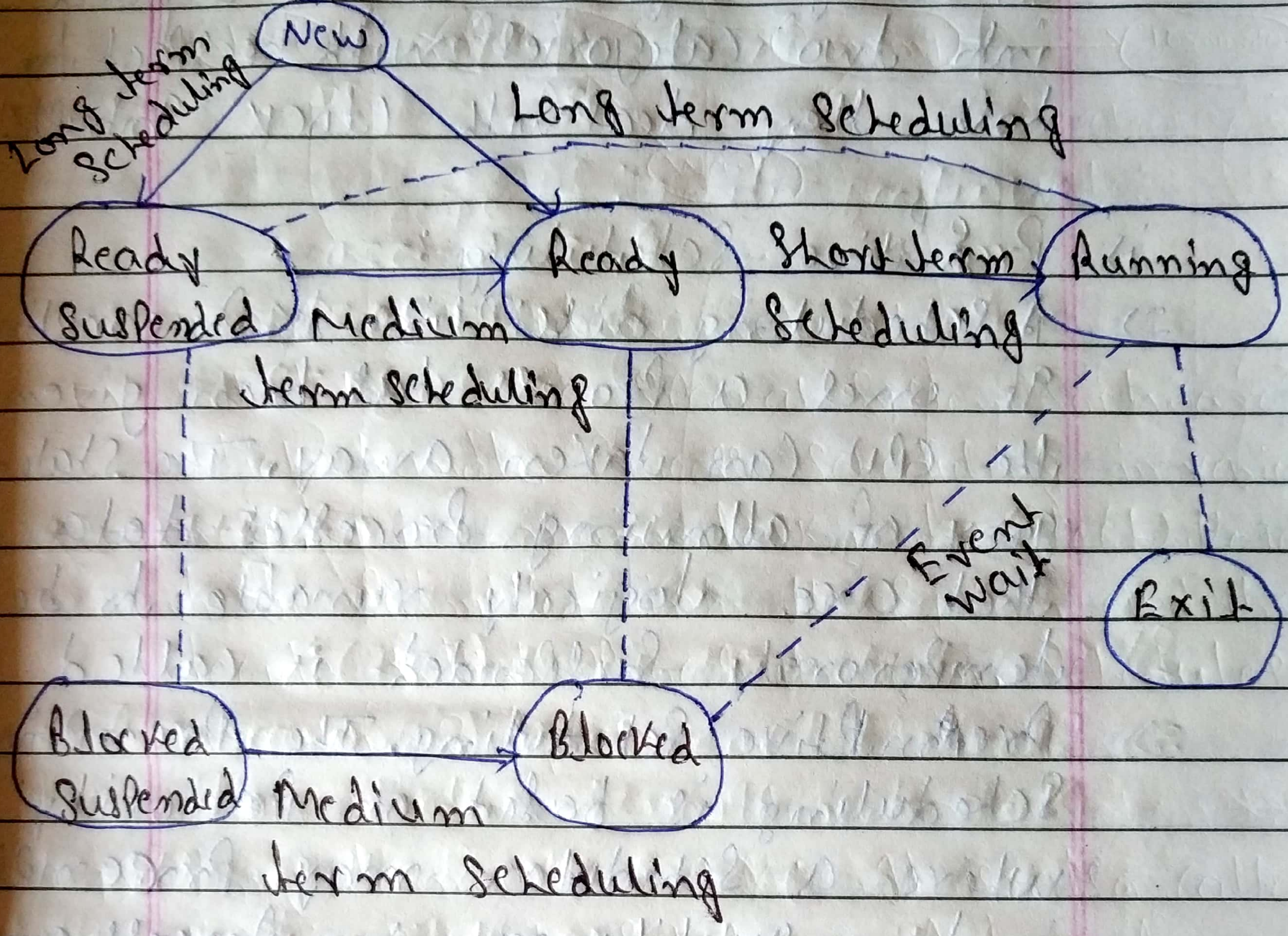
is called Swapping.



TOPIC :- Scheduling.

⇒ Scheduling refers to a set of Policies and mechanism supported by OS that Control the order in which the work to be done is completed. A Schedular is a operating System Program that select the next Job to be admitted for execution. The scheduling algorithm can be divided into two categories with respect to how they deal with clock interrupts.

i) Preemptive Scheduling :-

=> A scheduling discipline is preemptive if ones a process has been given the CPU can taken away. The strategy of allowing processes ~~data~~ that are logicaly runable to be temperoraly suspended is called preemptive scheduling. The premptive scheduling is based on periority where a schedular may preempt in low quality running process any time when a high periority process enters into a ready state.

ii) Non-Premptive Scheduling :-

=> Non-Premptive algorithm are design so that ones a process enters the running state it can not be premited until it completes its alloted time.

Topic:- Scheduling Algorithm.

1) FCFS (First-Come-First-Served):-

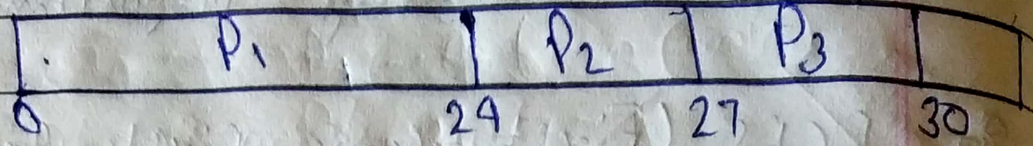
⇒ FCFS is the simplest scheduling algorithm. Its implementation is straight forward which is maintained by FIFO (First-in-first-out) queue. Once a Process has the CPU, It runs to completion. A FCFS scheduling is non preemptive which usually results in poor performance. As a consequence of non preemption, there is a low rate of resource utilization and system throughput. Short Jobs may suffer considerable turnaround delays and waiting times when CPU has been allocated to longer Jobs.

Example:

Calculate the average waiting time for the given set of processes.

Process	Processing time
P_1	24
P_2	3
P_3	3

=) Grantt Chart:



Waiting time:

$P_1 = \overset{24-24}{\cancel{0-0}} = 0$

$P_2 = 27-3 = 24$

$P_3 = 30-3 = 27$

∴ Average waiting time = $\frac{0+24+27}{3} = \frac{51}{3} = 17$

2) Shortest-Job-first (SJF):-

=) Shortest-Job-first is a non-Preemptive discipline in which waiting Job (or Process) with the smallest estimated run-time-to-completion is run next. In other words, when CPU is available, It is assigned to the Process that has smallest next CPU burst. If two Processes have the same CPU burst, FCFS is used. The SJF scheduler searches the ready queue to find the Job or the Process with the shortest execution time. SJF scheduling is an optimal scheduling algorithm in terms of minimizing the average waiting time of a given set of Processes

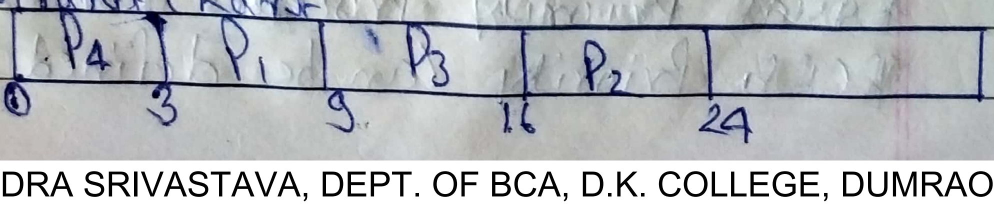
The Optimal Performance of SJF Scheduling is dependent upon future Knowledge of the Process/Job behaviour. This comes on a way to effective implementation of SJF Scheduling in Practice, because there is difficulty in estimating future Process behaviour reliable except for every Specialised deterministic cases. The major drawback of the SJF Scheduling is the Starvation because longer Processes are indefinitely PostPoned by the Smaller Process. Like FCFS, SJF is non Preemptive therefore, It is not useful in timesharing environment in which reasonable response time must be guaranteed

Example:

Calculate the average waiting time for the given Set of Process. which arrived at the Same time

Process	Processing time
P_1	6
P_2	8
P_3	7
P_4	3

Grantt chart

=>  waiting time:

$P_4 - 3-3 = 0$

$P_1 = 9-6 = 3$

$P_3 = 16-7 = 9$

$P_2 = 24-8 = 16$

$\therefore$ Average waiting time $= \frac{0+3+9+16}{4} = \frac{\cancel{28}^{7}}{\cancel{4}}$

$= 7$ mili sec.

3) Priority Based or Event-Driven Scheduling:-

⇒ A Priority number (integer) is associated with each Process. The basic idea is straightforward: each Process is assigned a Priority, and Priority is allowed to run. Equal-Priority Processes are Scheduled in FCFS order. The Shortest-Job-First (SJF) algorithm is a special case of general Priority Scheduling algorithm. SJF is a Priority Scheduling where Priority is the Predicted next CPU burst time. The CPU is allocated to the Process with the highest Priority (Smallest ~~for~~ integer = highest Priority). Equal Priority Processes are Scheduled FCFS. Priority can be defined either