

Name = Farhan Akhtar

ID = 15554

Subject = Operating System

Semester = 3rd

Teacher = Sir-Daud-Khan

Degree = BS(S.E)

Ques. ②

Ans :- Deadlock with one process is not possible.

Here is its explanation:

A deadlock situation can arise if the following four conditions hold simultaneously in a system

- mutual exclusion
- Hold and wait
- No preemption
- Circular-wait

It is not possible to have a circular wait with only one process thus failing a necessary condition for circular wait. There is no second process to form a circle with the first one. So it is not possible to have a deadlock involving only one process.

Q no 3 (3) :-

Ans :- Suppose the System is deadlocked
This implies that each process is
holding one resource and is waiting
for one more. Since there are three
processes and four resources one
process must be able to obtain two
resources. This process requires no
more resources and therefore it will
return its resources when Done.

Q no 5 (5)

Ans :-

Detection of starvation requires
future knowledge since no amount
of record keeping statistics on processes
can determine if it is making
'progress' or not. However, starvation
can be prevented by 'using' a
process. This means maintaining a rollback
count for each process and including
this as part of the cost factor in
the selection.
Process for a victim for
preemption / roll back.

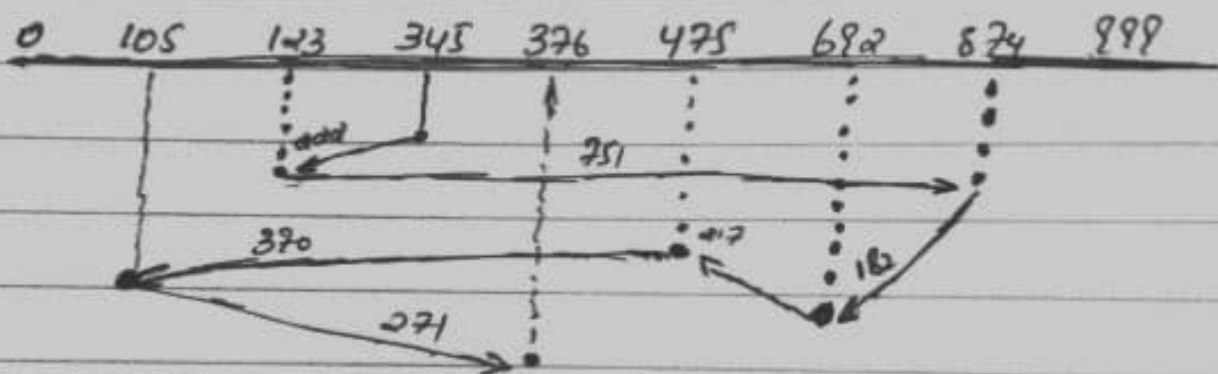
Q no 6:- (6)

Ans:-

Disk Scheduling Algorithms:

①:- FCFS:-

FCFS is the simplest of all the disk scheduling algorithms. In FCFS, the requests are addressed in the order they arrive in the disk queue.



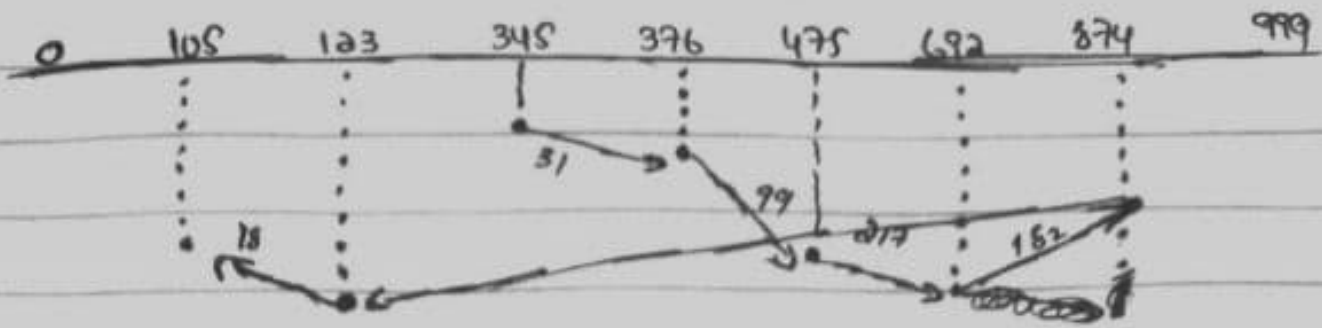
Total R/W Head movement:-

$$222 + 31 + 199 + 217 + 182 + 125 = 1076$$

②:-

SSTF:-

In SSTF (Shortest Seek Time First) request having shortest seek time ~~of every~~ ~~request~~ are executed first. So the seek time of every request is calculated in advance ~~in~~ in queue and then they are scheduled ~~and~~ according to their calculated seek time.



Total Row Head movement =

$$31 + 99 + 182 + 217 + 182 + 751 + 18 = 1298$$

Q no 1 — ①

Ans :- Deadlock :-

Deadlock is a situation where a set of processes are blocked because each process is holding a resource and waiting for an other resource acquired by some other process.

Consider an example :- when two trains are coming toward each other on same track and there is only one track none of the train can move once they are in front of each other. Similarly situation occurs in operating system.

Necessary Condition :-

There are four conditions that are necessary to achieve deadlock.

(i) Mutual Exclusion:
At least one resource must be held in a non shareable mode: if any other process requests their resources, then that process must wait for the resource to be released.

(ii) Hold & wait: A process must be simultaneously holding at least one resource and waiting for at least one resource that is currently being held by some other process.

(iii) No preemption:
Once a process is holding a resource (i.e. once its request has been granted), ~~the one its request had~~ then that resource cannot be taken away from that process until the process voluntarily releases it.

(iv) Circular wait:

A set of processes $\{P_0, P_1, P_2, \dots, P_{N-1}\}$ must exist such that every P_i is waiting for $P_{(i+1) \% N}$ (note that this condition implies the hold and wait condition).

Qno (4) - (4)

Ans:- Resource Allocation Graph:-

The Resource Allocation Graph is the pictorial representation of the state of the System. As its name suggest, the Resource Allocation Graph is the complete information about all the processes which are holding some resource are waiting for some resource.

It also contains the information about the all the instances of all resources whether they are available or being used by the processes.

In Resource Allocation Graph the process is represented by a circle while the Resource is represented by a rectangle. Vertices: vertices are mainly of two types Resource and process each of them will be represented by a different shape. circle represent process while rectangle represent resource.

Edges: Edges are also of two types, one represent assignments and other represent the wait of process for a resource. The above image each show of them.

A process is shown as waiting for resource if the tail of an arrow is attached to the process.