

Subject \rightarrow Operating System
Assignment - 4 \rightarrow

Part - A : Short Answer Type \rightarrow
From Q. (1 to 5) similar to Assignment
Three From Q. (1 to 5).

Part - B : Application / Numerical Based \rightarrow
Distributed Deadlock Detection \rightarrow

(a) Global Wait-For Graph: From all sites combined:
 $P_1 \rightarrow P_2, P_3 \rightarrow P_4, P_2 \rightarrow P_5, P_5 \rightarrow P_6, P_6 \rightarrow P_1$

(b) Deadlock Detection \rightarrow There is a cycle:
 $P_1 \rightarrow P_2 \rightarrow P_5 \rightarrow P_6 \rightarrow P_1$.

Thus, a deadlock exists involving P_1, P_2, P_5 and P_6 .
Suitable distributed algorithm \rightarrow

The Chandy - Misra - Maas algorithm can be used,
as it detects cycles in distributed wait-for
graphs with minimal communication overhead.

(7) Distributed File System Performance \rightarrow

(a) Expected access time:

$$\begin{aligned} \text{Expected time} &= (\text{Probability Local} \times \text{Local time}) \\ &+ (\text{Probability remote} \times \text{Remote time}) \\ &= (0.7 \times 5\text{ms}) + (0.3 \times 25\text{ms}) \\ &= 3.5\text{ms} + 7.5\text{ms} = 11\text{ms} \end{aligned}$$

(b) Caching Strategy \rightarrow

A client-side caching strategy with consistency
checks can improve performance of frequently accessed
remote files are stored locally, reducing remote
access and lowering average access time while
maintaining correctness.

(8) Checkpointing Strategy :->

Full checkpoint = 200ms, Incremental checkpoint = 50ms
 RPO = 1 second, Period = 10 seconds

(a) Optimal mix :

Use one full checkpoint at the beginning and then incremental checkpoints every 1 second.

This results in: 1 full + 9 incremental checkpoints.

(b) Reasoning :

The incremental checkpoints ensure that no more than 1 second of work is lost, satisfying the RPO. The simple full checkpoint provides a complete task state baseline, while incremental checkpoints minimize overhead across remaining time.

(9)

Global E-Commerce Platform Case Study ->

(a) Distributed scheduling challenges & suitable algorithm ->

During flash sales, request surges cause uneven load across servers, unpredictable arrival patterns, and potential bottlenecks due to geographic distribution.

A suitable algorithm is Dynamic Load Balancing using Least-Loaded or Work-Stealing approach, which continuously redistributes tasks based on real-time server load, ensuring efficient utilization.

(b) Fault tolerance strategy with RTO & RPO considerations ->

A combination of geo-redundancy and synchronous replication ensures minimal data loss (low RPO). Services are replicated across multiple regions so that if one data center fails, another immediately takes over. For fast recovery (low RTO), implement automatic failover through distributed coordination services. This ensures continuous availability even during regional outages.