Prerna Kanapar
2301010251
Btech CSG - 'D'

Date ..../..../.....

Assignment 04

Ques 1. To form the global wait-for graph (combined) :-

given :-

from all 3 sites local edges

S_1 $P_1 \rightarrow P_2$, $P_3 \rightarrow P_4$

S_2 $P_2 \rightarrow P_5$, $P_5 \rightarrow P_6$

S_3 $P_6 \rightarrow P_1$

graph :-

$P_1 \rightarrow P_2 \rightarrow P_5 \rightarrow P_6$ (cycle)

$P_3 \rightarrow P_4$ (not part of cycle)

(b) There is a cycle.

$P_1 \rightarrow P_2 \rightarrow P_5 \rightarrow P_6 \rightarrow P_1$

Hence, deadlock exists and processes involved P_1, P_2, P_5, P_6. Processes P_3 & P_4 are not part of the deadlock.

(c) Chandy-Misra-Has probe-based deadlock detection algorithm is a suitable algorithm :-

→ Each site sends a probe message along wait-for edges.

→ If a probe returns to its origin, a cycle is detected.

→ Works well in distributed environments where no single site has the entire graph

Spiral

Date ..../..../.....

Ques2 Expected access time = 0.3 x 25 ms + 0.7 x 5 ms
=> 7.5 + 3.5
=> 11 ms
One caching strategy to improve performance is popular files are served locally. Leases keep caches reasonably consistent without excessive invalidation traffic giving a good tradeoff b/w latency and correctness.

Ques3 (a) Recommended Schedule (over 10s):
→ If a recent full checkpoint exists before the 10s window: take only incremental checkpoints every 1 second (10 incremental checkpoints).
→ If no prior full exists: take one full checkpoint at the start, then incremental checkpoints every 1 second (1 full + 10 increments across the period).
Overhead calc:-
with prior full = 10 x 50 = 500 ms
without prior full = 200 + 10 x 50 = 700

b) RPO → 1s means you must capture changes at least every second. Incremental checkpoints are much cheaper (50 ms) so scheduling them every second minimizes overhead while meeting RPO. A full checkpoints is only needed occasionally to bound restore complexity — otherwise pure incrementals plus an earlier full give the lowest cost.

Spiral

Date ..../..../.....

Ques 4 * Scheduling Challenges are :-
sudden traffic spikes, hot product partitions, cross-region latency, uneven load, data locality vs consistency tradeoffs.
Suggested algorithm uses power-of-d-choices (P2C) front-end routing combined with Rendezvous / consistent hashing for session affinity & data locality; P2C gives very low queuing imbalance, consistent hashing routes requests user data to reduce latency.

* Fault tolerance strategy includes :-
→ deploy active-active multi-region services
→ critical metadata / state.
→ Bulk data / catalogs
→ using periodic full + incremental backups
→ For RTO :-
automate failover and warm standbys so services in another failover region take traffic in seconds.
For RPO :-
ensure critical data is synchronously replicated or use cross-region commit protocols.

Sirat
21/4/25