Lab 7

Q1

No, latency cannot be eliminated completely because it is an inherent property of any system that involves communication, processing, and data transfer. Latency is influenced by several factors, including:

1 Speed of Light Limitation: Information travels at the speed of light, which, while extremely fast, is still a finite speed. Any physical distance between two points will result in a delay

2 Processing Delays: Even if the transmission itself were instantaneous, there are delays associated with processing the information at each end. This includes queuing, routing, and application-level processing

3 Physical Distance: As networks span larger geographic areas, the distance the data must travel increases, directly impacting latency

4 Network Congestion: Network congestion can cause delays as data packets are queued or retransmitted.

5 Hardware and Software Limitations: Hardware and software processing times contribute to latency

Q2

Latency and consistency are closely related in distributed systems and multiplayer games. High latency can lead to inconsistencies between different clients' views of the system state

Example: Shooting Around Corners (Multiplayer Game Scenario)

Imagine a first-person shooter (FPS) game where Player A is hiding behind a wall while Player B is shooting at them

Latency Effect:

Player B sees Player A still exposed and shoots

Due to network delay, Player A has already moved behind the wall on their screen.

The server receives the shot and decides whether Player A is hit based on the time at which the server received the data

Consistency Issue:

If the game favors Player B (low latency to server), Player A may get hit even though they were behind cover

If the game favors Player A (by applying lag compensation), Player B’s shot might "miss" despite appearing accurate on their screen

This example illustrates how higher latency can cause inconsistencies between different players' views of the game world, affecting fairness and gameplay

Q3

Dumb Client (Conservative Technique)

Definition: A dumb client relies entirely on the server to process and validate actions. The client only receives updates from the server without making local predictions

Category: Conservative consistency management

Pros: Ensures a high level of consistency since the server dictates the final state.

Cons: Introduces visible lag for the player, as every action requires a round-trip to the server.

Client-Side Prediction (Optimistic Technique)

Definition: The client makes local predictions about the outcome of an action before receiving confirmation from the server. If the prediction is incorrect, the game state is corrected.

Category: Optimistic consistency management.

Pros: Reduces perceived latency and provides smoother gameplay.

Cons: Can lead to temporary inconsistencies (e.g., a player sees themselves moving forward but gets "snapped back" if the server disagrees).