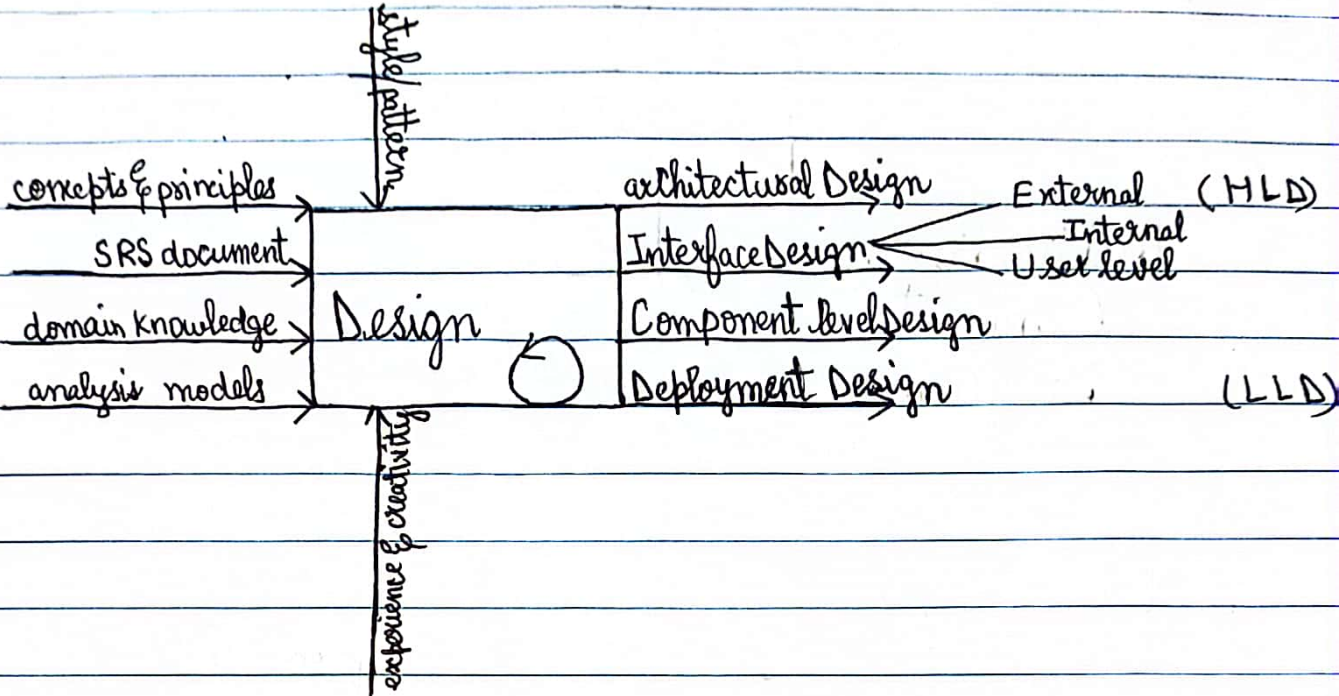




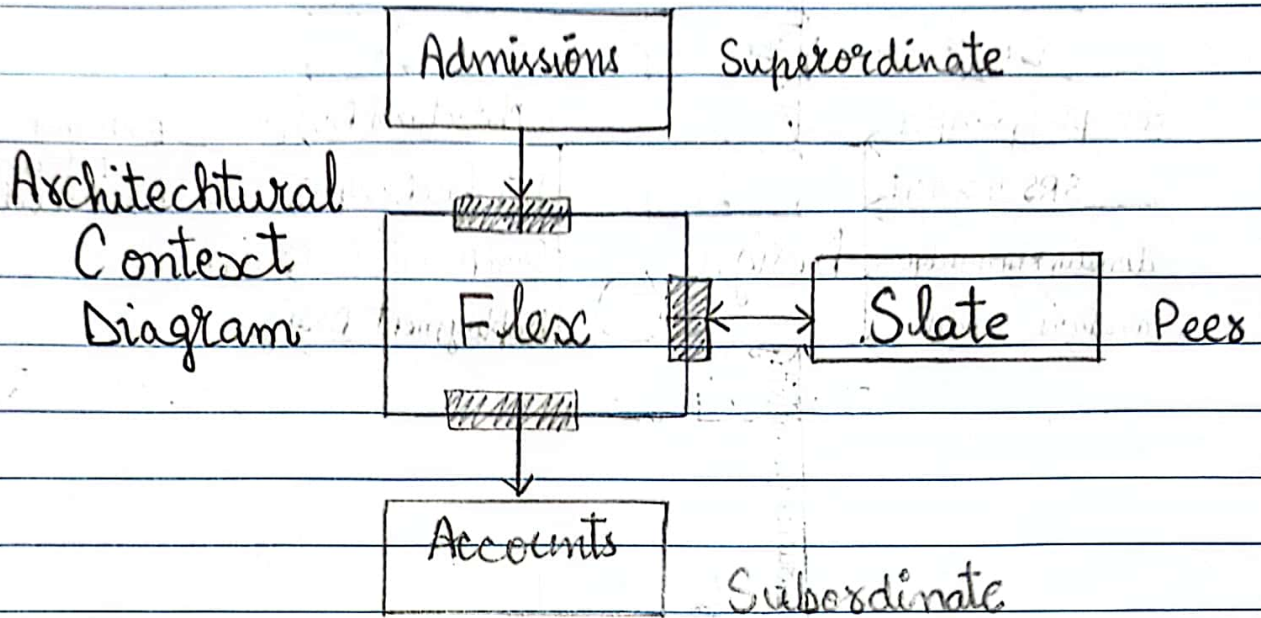
Design



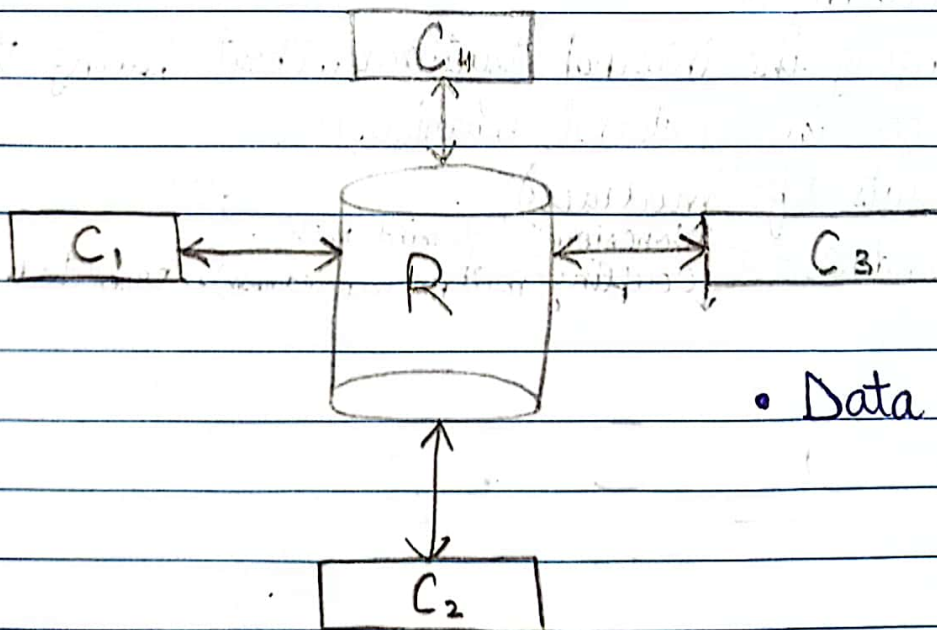
- Abstraction (selective selection) :)
- Refining :)
- Information Hiding :)
- Refracting (changing the internal structural without having the change in external behaviour).
- Modularity (consists of modules)
- Functional Independence
 - ↑ Cohesion (single mindedness)
 - ↓ Coupling (multiple mindedness), (connectedness)



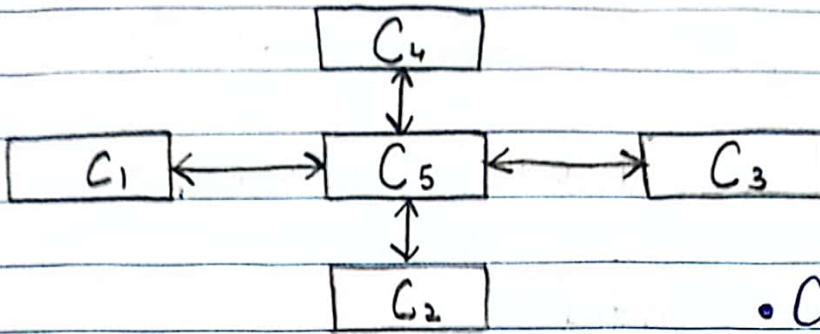
Software Architecture (aka HLD)



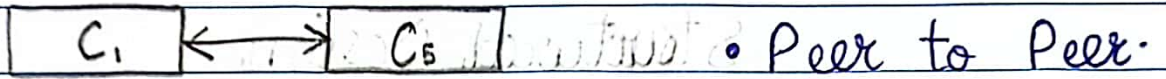
Architectural Styles



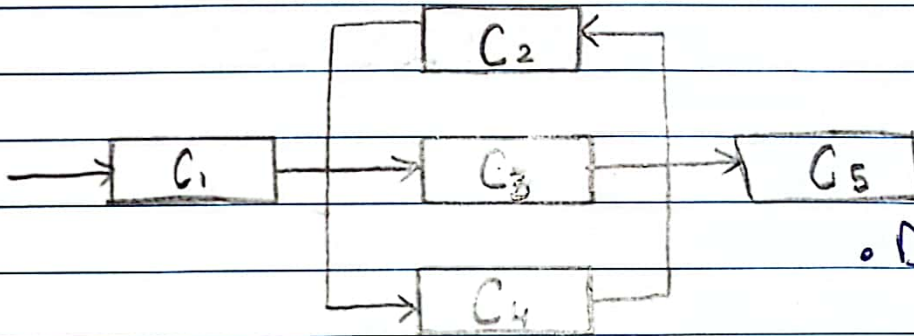
• Data Centered



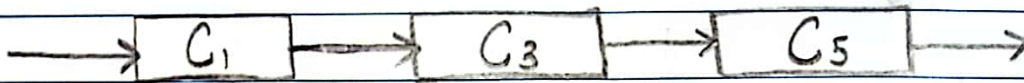
• Client Server.



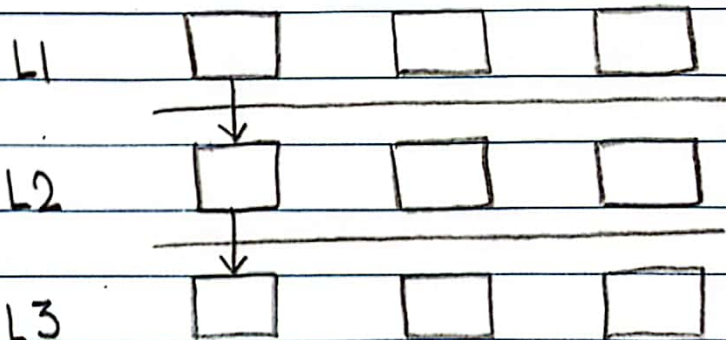
• Peer to Peer.



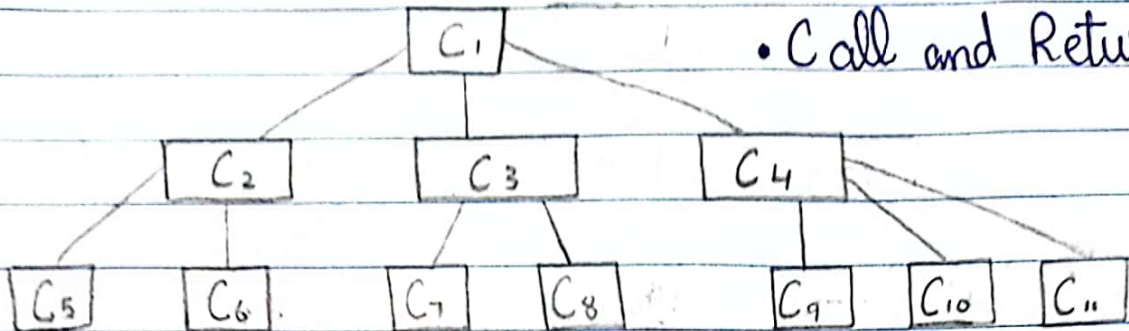
• Data Flow



• Batch Sequential



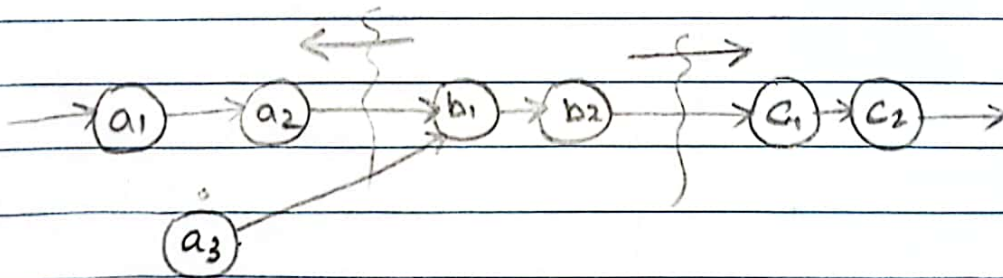
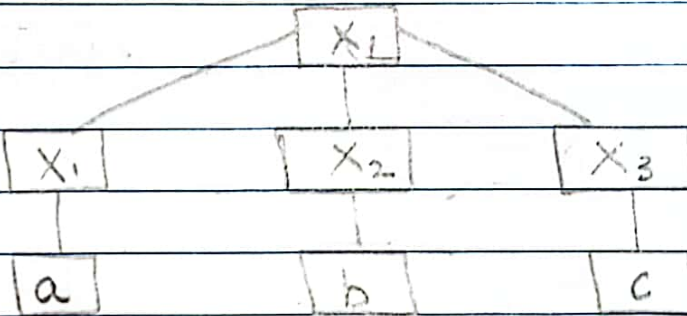
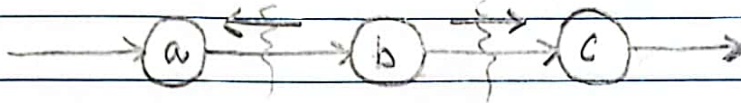
• Layered/Tiered

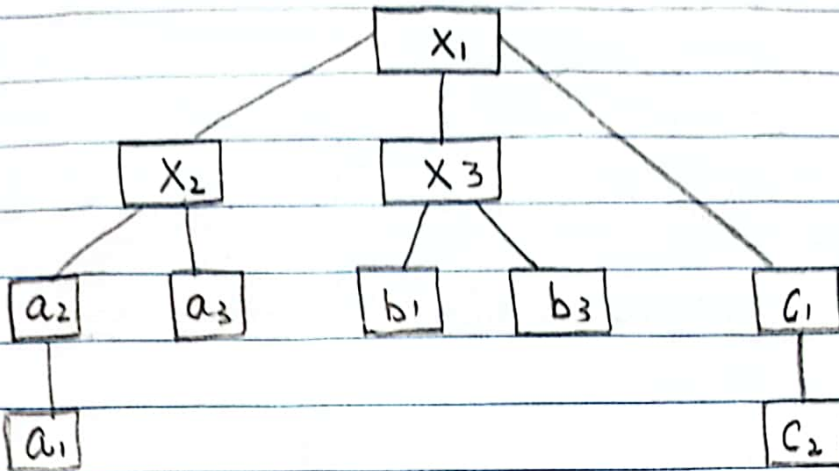


• Call and Return.

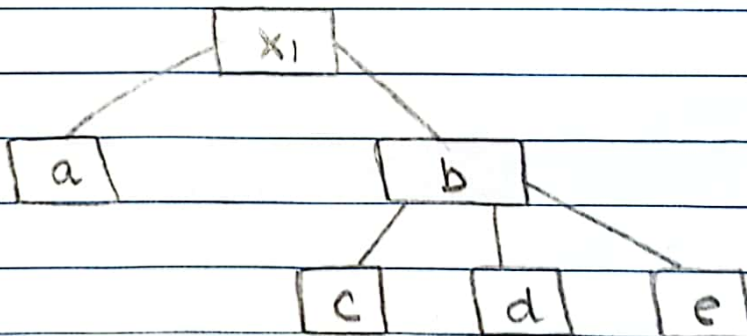
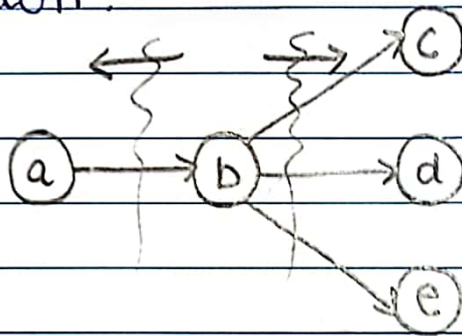
Structured Design

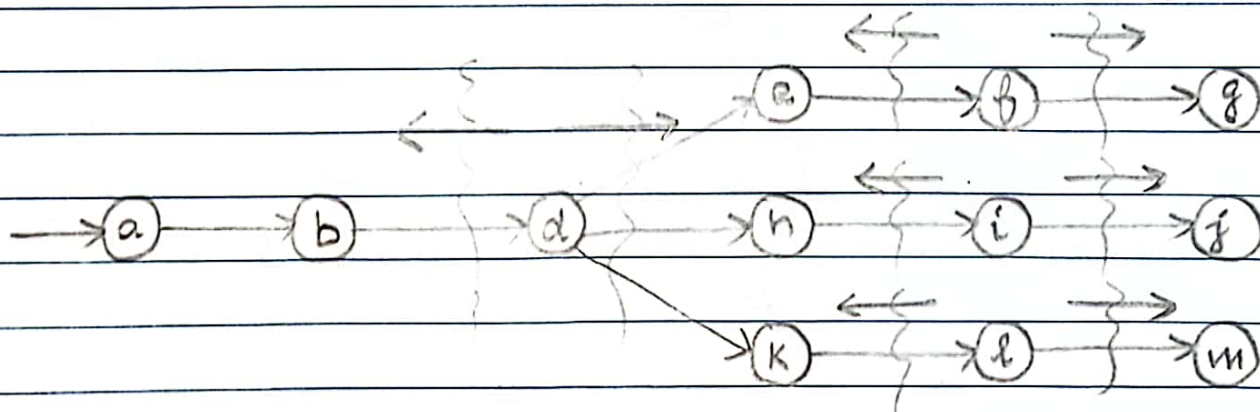
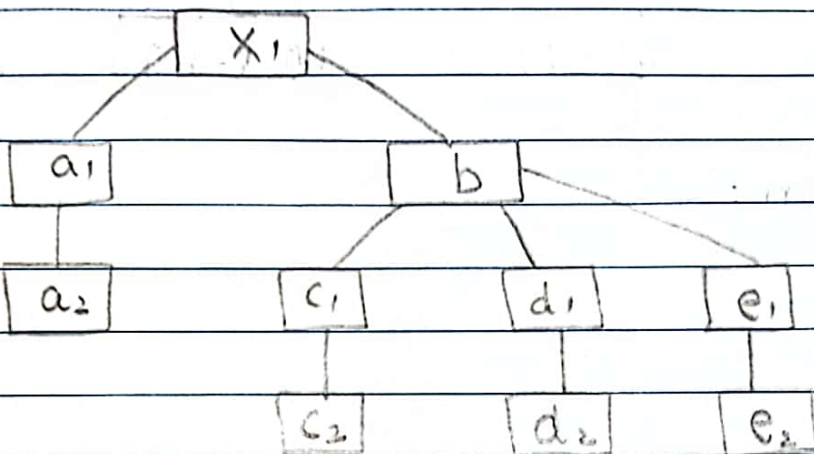
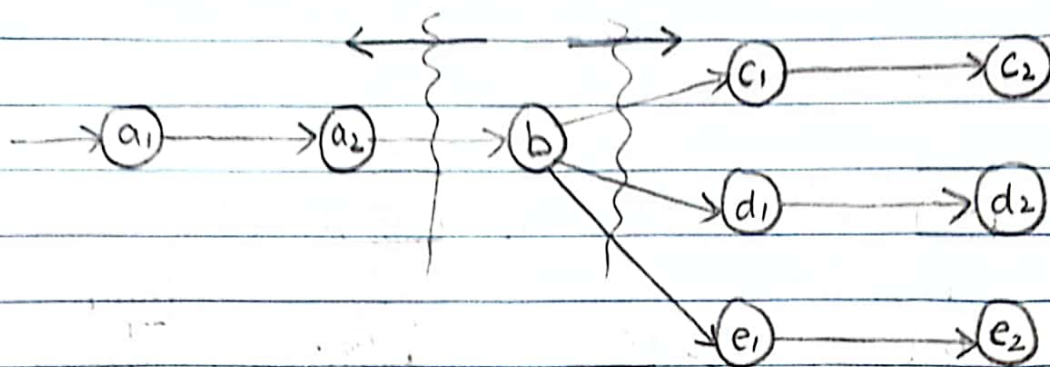
1) Transform:





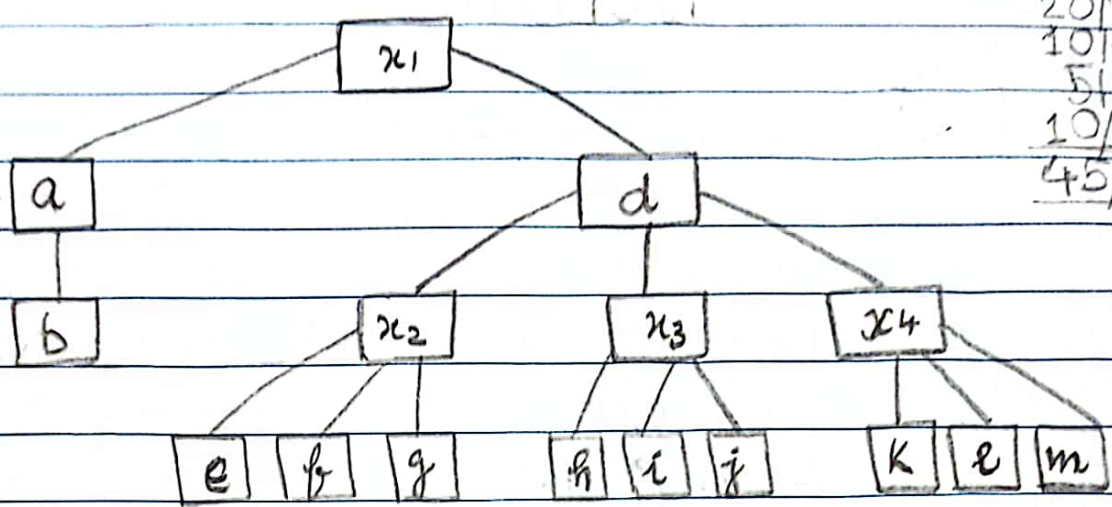
2) Transaction:







20/25
10/15
5/20
10/40
45/100



Golden Rules

- Give more control to user.
- Have less memory load (recognition vs recall).
- Consistent (global used features/hints).
- User dependent interfaces

Analysis

- user (types)
- task (work flow)
- environment (physical condition of user and system).

Design

- Font size, Font Style, Text colour, Text format.
 - Colours, gradients
- Implementation → Evaluation.

