

Name: <Haitham Babbili>

Personal number: <19880803-4915>

### Question 1:

1.a.i : The traffic control more efficient, you don't need to modify every routing table in every device just one time for one device and it will help to centralize the decision and management of the network.

1.a.ii : it increases the complexity and the device cost since the switch needs to ask the controller every time about routing rules and keep querying to know the topology.

1.a.iii : we will lose security control on the information since we remove the routing table from our side which means we will lose the firewall on this device and the network always under attack threat

1.b.i : no since transport layer is run at the end system and does not send any more info until it gets acknowledgment

1.b.ii : usually TCP/IP is more reliable system. The data goes through transport layer then network layer to physical layer, the data sent with IP and port number. First the client sends synch to the server, server listening if free then sends acknowledgment to client, client sends packet synch ack back to inform the server it's synch now, the client sends the message and waits for acknowledgment from the server to send next message, in failure case the client sends the message again

1.b.iii : there is no contradiction since TCP/IP will work to eliminate the congestion caused by sending huge data rate in fast way by controlling the flow in the network and delivering the data, Data link algorithm is also work to deliver the data and eliminate the congestion caused by sender or receiver crash.

1.c.i : that means there is no record to DNS which means the local server will ask the root server about the domain where if the domain is unique the root will give it otherwise it will be hard to know the domain since there is no record and will ask every domain if he is the one which can be hard process and maybe we get wrong DNS since there is no record and no type of the DNS (A record or NS, IPV4).

1.c.ii : If DNS cache for long time will cause memory full for the DNS server which means work slowly because the long time of search and need more memory.

1.d.i : to self-stabilize network: the network should be synchronous and has limit or bound i.e the network should be known, if it is anonymous means it's not known, no limit or boundary so no synchronous

1.d.ii : by converting shared memory to message passing to use compiler that done by using token passing algorithm where only one processor holds the token and sender and receiver can hold it for infinitesimal time, until new token comes. The counter in network will use to avoid deadlock configuration. If sender has message greater than counter, he will set the counter to his message +1, and send its counter message to the receiver where the receiver will compare it with his own counter if it's not equal, set its counter to equal new counter

message it received. By using Token passing algorithm will arrive to safe configuration then the system will stabilize.

**1.e.i :** when it used super self -stabilizing , since it reduce number of rounds in network topologie

**1.e.ii :** No we should not, since at the end it give use the behaviour we want and we designed to, But we can use a dynamic system in case of tolerance or failure happened.