



Date: 8 - 05 - 2023

Time: 2.30 PM – 5.30 PM

Max. Marks: 50

N.B.: Answer ALL questions

All programs/functions/code segments are to be written in C/C++.

1. List functions of all the layers of ISO's OSI reference Model. What are all the protocols at various layers of TCP/IP which meet the corresponding layers functionality of OSI reference model? What is the 5-tuple in a socket? By what socket system calls, the 5-tuple gets associated with a socket connection? What is the major system call for multiplexing in network programming and give its syntax of usage? Which of the TCP/IP layers implementation requires multiplexing and support your answer with details. 10
2. What are all different types of process communication? Outline Remote Procedure Call (RPC) mechanism by showing the steps in a diagram. How many files are involved in Sun RPC mechanism and give their functionality? Which of these should be written by the programmer and which will be generated by rpcgen compiler? What are the limitations of RPC? What is the need of Name Server? 10
3. Write short notes on: (write answers for all sub questions at one place) 10
 - a) File Transfer Protocol (FTP) : Differentiate Active File Transfer Protocol (FTP) versus passive FTP? Why there is a need of two TCP connections in FTP? Support with appropriate port numbers that are gets exchanged between FTP server and FTP client. (Assume that many clients will be connected to a server and a client will download more than one file at a time.)
 - b) Internet routing algorithms/methods/protocols
 - c) Domain Names and Domain Name Service (DNS)
4. **NITW NetMeet** 10The following scenario of Networked Computers with the networked processes provides an online meeting feature.
 - * Process Organizer (O) opens a point of joining the meet. Once Participants (P) join in the meet, then O handovers the meet control to two Speakers (Sp). Process O also receives chat messages on this point of meet from any P and puts them in chat box. Once both the speakers send the message of completion to process O, then O sends a participation certificate to all those Ps who have sent at least one chat message and closes the meet.
 - * Process Speaker(Sp1) and Process Speaker(Sp2) speaks alternatively by sending signals among themselves. Each of them know the name of others(i.e. executable program name). When Sp1 is speaking, Sp2 will look at chat messages and answers. If a query in chat box is general, then Sp2 answers to it by putting the answer in chat box itself. If a query in chat box is expecting a personal answer by giving the contact point($1024 \leq \text{number} \leq 65535$), then by contacting to that P with that contact point, Sp2 will be sending(connection-oriented) the answers personally. When Sp2 is speaking, Sp1 will also take care of the chat box in the same way.
 - * Process Participant(P) first joins the meet and can listen(receive) to the speakers, and can also send chat messages. If any P wishes to get reply personally, he/ she can give his contact point and can wait for answer on that point. Once any one of the speaker contacts, then the P can accept and can receive the answer.

- * The processes O , Sp1 and Sp2 are in the same computer system. Each of the participant process P is in a different computer system.

First draw clear diagram of Networked Process Diagram (NPD) for the above scenario.

Then write full code for processes O , Sp, and P .

Clearly comment before every block of specific flow of logic.

(No need of writing #include statements and address initialization statements. No need of writing statements of repeated logic)

5. **NITW Online-Exam**

10

The following scenario of Networked Computers with the networked processes provides an online examination feature.

* Candidates appearing for online examination, have to execute two programs EO.exe and A.exe on their computer systems which are supplied by the Examination Controller(EC). The process Exam Observer (EO) runs in the background and the process Answering Screen(A) runs on foreground with full screen mode. After getting connected to EC , the first message from A should be the hall ticket number of the candidate which will be typed by the candidate.

* A candidate has to type answers for the questions appearing in the full screen of process A. The candidate should not open/use any other programs/browsers/windows/search tools. Any violation of this results in as: first two warning messages of violation will be signaled from process EO locally. Still, after receiving the two warning messages, if the candidate violates for the third time, then a violation of rules message along with the sites opened and cancellation of exam message from process EC will be received. And internet for the system will be stopped by the process EO as per the order from the EC process till the time of end of the exam. Assume that the candidate's computer system is on a LAN.

* The process EO observes the internet usage of the candidate's computer system and reports its findings to the process EC (i.e. about what are all the connections and services that are getting used). As soon as EO finds a new observation other than the communication between process A and the process EC, it signals process A, about the violation for two times. And for the third time violation, it sends a violation report to process EC. After that, it receives the order of Internet stopping to the candidate's system, and arranges the disconnection of Internet, by sending appropriate socket system calls to local LAN server.

* The process EC arranges for the conduction of examination for all the candidates through a well-known connection oriented service point. It also opens a general computer level connection with all the EO processes. The EC sends the question paper to all the connected A processes. After that EC starts receiving answers from A and stores them along with the concerned hall ticket number. EC also receives the violation reports from EO processes and acts as mentioned in the descriptions of process A and EO. The EC sends successful completion message to all the A processes who have not violated any rule till the end of the exam.

* The processes A and EO are in the same computer system(i.e. Candidate's computer).

The process EC is in a different computer system which is not on the same LAN of Candidate's computer.

First draw clear diagram of Networked Process Diagram (NPD) for the above scenario.

Then write full code for processes EC , EO , and A .

Clearly comment before every block of specific flow of logic.

(No need of writing #include statements and address initialization statements.

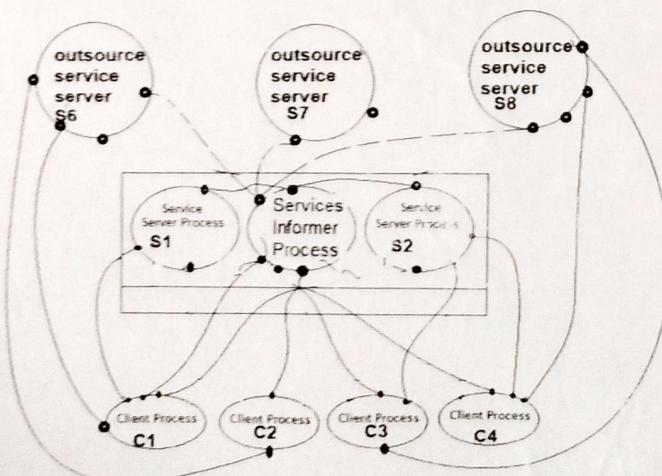
No need of writing statements of repeated logic)



Date: 20 - 04 - 2023 Time: One Hour Max. Marks: 10

Note: Answer all questions with subparts in sequence at one place.

- 1.a) What are the ranges (values) of source port and destination port numbers in the first message of a TCP connection establishment in the Internet? How you have arrived at such ranges? 1
- b) Which fields of the IP header change from router to router and why? 1.5
- c) Is there need to have total length field in TCP Header, which is present in UDP header? 1.5
First answer 'Yes' or 'No'. After that only briefly explain to support your answer.
- d) Router A has to send a packet of size 250 bytes to Routers B, C, and D which offered their acceptance for 110, 90, and 30 bytes respectively. What are the fragment sizes that A will send? How you have arrived at such sizes? 1.5
2. Consider the following figure. Client Processes send an enquiry message to Server computer system, asking for what are all the services available, at its place (in the same computer system) and at other places (other computer systems). The Services Informer process will only receive this message, replies to the clients about the services information it has got from inside Service server processes (same computer system, here in figure : S1, S2) and outsource service server processes (other computer systems, here in figure S6, S7, S8). And also the Services Informer process also provides one connection-oriented service S0 on its own through threads. In other words, whenever any (new) inside service server process or any (new) outsource service server process wishes to provide a service, it informs about that to Services Informer process through appropriate connections. After getting the information about services, a client connects to the required service sever process. A client can connect to multiple server processes i.e either to Service Informer Process for service S0 or to an inside Service Server Process (for services S1, S2) or to an outsource service server process (for services S6, S7, S8). Write sample(specific) code for Service Informer Process and for a Client process.
(No need of writing #include statements and address initialization statements.
No need to use any IPC mechanisms for this scenario.) 4.5



Scenario:
Inside Service server processes S1, S2, and outsource service server processes S6, S7, S8 have sent about their services to Service Informer Process.
C1, C2, C3 C4 got information from Service Informer Process.
C1 – using services S6, S1, and S0 service of Service Informer Process, and the Service Informer Process is providing S0 through a thread.
C2 : using service S6
C3 : using services S2, S8
C4 : using services S2, S8



Date: 25-02-2023

Time: 2.00 PM – 4.00 PM

Max. Marks: 30

N.B.: Answer ALL questions

All programs/functions/code segments are to be written in C/C++.

What is the need of seven layers in ISO's OSI Reference model? Elaborate on functions of first six layers bottom to top. What is the need of three addresses in computer networks? Can we use only two addresses? If yes, which of them, if not why? 7.5

S is a Super server process which supposes to offer four services (S1, S2, S3, S4) by listening to four socket fds. As soon as the first client connect request arrives for a service (say S3), {in other words, as soon as the super server process S notices the first request for a service}, the super server process S creates that particular service server process (say S3) by exec() the .exe file (S3.exe). From then onwards, that particular service server process (say process S3) only accepts all the client requests for that service and serves them using threads. Write pseudo-code (with proper sequence of system calls) for process S and also for any service server process Si. (Note that The Super Server S is not at all accepting the client connect requests) 7

3.

Proxy Server

Assume that Clients are not allowed to communicate directly with Special servers. The Clients request the Proxy Server for such facility by providing the Special server number. The Proxy connects to the corresponding Special server on behalf of the client and mediates the communication as Client \leftrightarrow Proxy \leftrightarrow Special server. The Proxy handles all connections of the Special server with a single separate thread i.e. five threads for five Special servers. Assume that Proxy knows the address details of all Special server processes and all processes are in different computer systems (see figure).
Proxy also provides two services on its own.

Implement Proxy, Special Servers and a Client process using BSD sockets

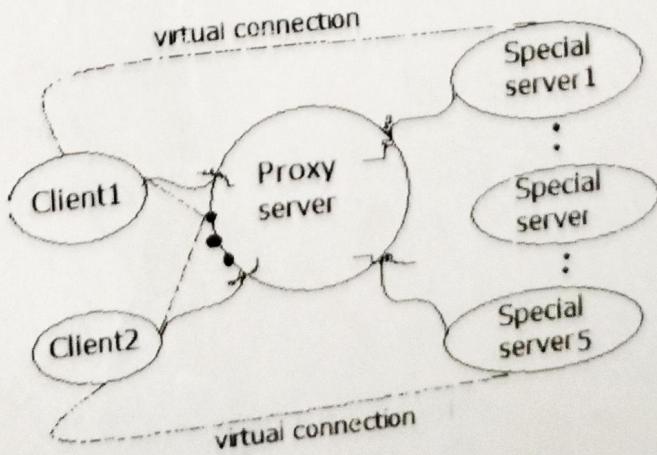


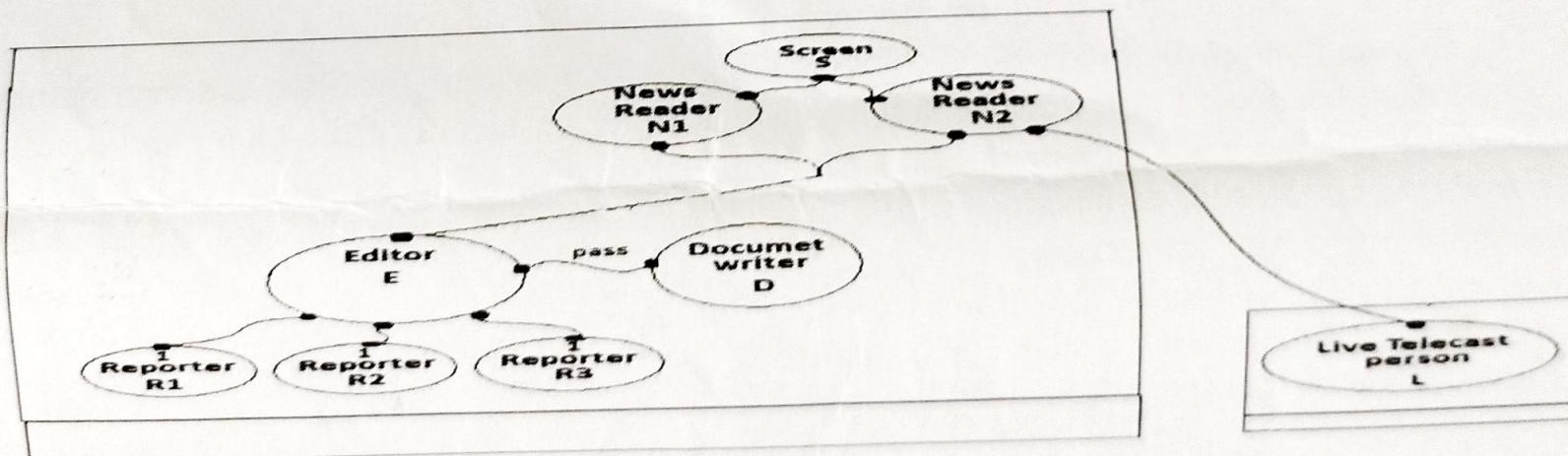
Figure: Proxy Server

4. Nitw-News

Consider the following different processes at Nitw-News Channel Studio as shown in sample figure below. Editor(E) gets news from Reporters(R). Reporters type news through keyboard and send to standard output, which in turn goes to Editor. Whenever Editor receives a news, he sends the news to News Reader's table. The News Readers N1 and N2 reads the news alternatively, i.e. arrange to display the news onto the Screen. (A news will be displayed only once onto the screen). If a News Reader notices a decimal number(port address) in the start of news content, then it gets connected to a Live telecaster(L) whose IP address is IP₅, and the noticed port number. After connecting, the News Reader arranges to echo onto the screen, whatever it gets from Live telecaster. At this moment the other News Reader should not read any news as live telecast is going on. Once live telecast is over, the news reading will be done alternatively by N1 and N2 as usual. The News Reader who has got connected to live telecast should also notify the Editor, so that the editor will maintain a total count of live telecasts at any moment. This count can be seen/known by anybody in the studio. If Editor notices a news message with prefix "/d" from any of the Reporters, then it passes that Reporter to Document Writer(D). The Document Writer stores the news messages that it gets from the Reporter in a file(document).

First mention clear steps and then write Pseudo-code/ Sample Code for all the different type of processes involved in this News Channel Studio scenario.

You must have to use all the IPC (Inter Process Communication) mechanisms, **90%** of I/O **multiplexing**, required BSD socket system calls and other suitable system calls.



Date 23-02-2023

Time: One hour

Max. Marks: 10

1. Give the list of application layer protocols of TCP/IP along with their functionality. 2
 2. Process P1 has to facilitate (arrange) that the standard outputs (as and when available) of processes P2,P3, and P4 should go as standard input of process P1. Then, the standard output of process P1 has to go as standard input of process P5. For example, in process P3 executes cout<< "CSE NITW born to think and best to win '\n'; then process P5 should be able to display "CSE NITW born to think and best to win" in its window(screen) by executing the statements cin>>bus; cout<<bus; 3
- Write pseudo-code (with proper sequence of system calls) of Process P1 only.
 (You should not use read(), write(), select() system calls and also threads)
3. A server process S gets input data from processes P1, P2, P3 and from keyboard (standard input). P1 is connected through pipe and P2 is connected through FIFO(named pipe) to S. Process P3 sends its standard output to S. A child Process S' is also gets communications from server S. If S gets data from keyboard or P1 or P2 or P3, it sends the same data to child process S'. If S gets a signal from process P4, it puts the data which it receives (soon after the signal received), from keyboard or P1 or P2 or P3 as a message into the message queue. That means all the messages in the message queue are the ones which were that of soon after receiving a signal from P4. 5
- Write a program (full code) for the server process S only. The program should not use threads and poll().
- (No need of writing programs for P1, P2, P3, P4).

