Nepal College of Information Technology

Subject: Network Programming (BESE VIII)

Tutorial

Tutorial submission is compulsory.

1. Why UDP is faster than TCP? Which function is responsible for sending SYN segment during TCP connection establishment phase? Illustrate the TCP 3-way and TCP 4-way handshake mechanism with suitable state transition diagram.
2. How network programming is different from computer network? Explain different protocols (at least 3) in each layer of OSI reference model. Compare TCP, UDP and SCTP.
3. *While establishing TCP connection, the initial sequence number should not start from 0. Why? What is 2MSL wait state?*
4. What are the ways to pass the length of socket structure for different socket API's argument? Explain them in detail with function prototype and argument detail.
5. Explain the statement: fork () is called once and it returns twice. What is the significance of calling exec() after fork() system call? What will happen when close () is called in TCP socket during the implementation of concurrent server?
6. What is socket? Explain different system call in specific order, required to create a TCP client and TCP server in the Unix System. (Hint: Explain each function with prototypes, return types and sample code)
7. What is the purpose of bind () function? What will be the outcomes if we do not specify IP address, port, both, or neither during bind () system call?
8. What are the major differences of wait () and waitpid()? Explain the mechanism to handle multiple client in Unix network programing with suitable sample code (consider simple client server chat)
9. What is completed connection queue and incomplete connection queue ()? Explain different system call in specific order, required to create a UDP client and UDP server in the Unix System.
10. Can we use connect () system call while developing UDP socket? If yes, what will be the outcomes of using connect () in UDP socket? If no, how communicating processes know each other? Explain them in detail.
11. The connect () function returns only when the connection is established or an error occurs, what are those errors and when these errors occurred? Explain in detail.
12. What is listening socket and connected socket? Explain with function prototype and sample code: when and why getsockname() and getpeername() required?
13. Explain the functions send (), sendto (), recv () and recvfrom (). What are the major differences between wait () and waitpid()? Explain the possible option values that we can supply in waitpid() system call.
14. What is the role of socketpair() function? Explain different system call in specific order, required to create a UDP client and UDP server in the Unix System.
15. Differentiate unix domain socket and internet domain socket. Explain the mechanism of passing file descriptor in the Unix System.
16. What is signal and why it is required? Explain POSIX signal handling mechanism. Also Explain few signal types.
17. Why IO and IO multiplexing is needed in socket programming? Explain various IO models in Unix system.
18. What is I/O multiplexing? Explain the use of select function in the context of I/O multiplexing in detail. (Hint: Do not forget possibilities of wait options)
19. Differentiate blocking IO model with non-blocking IO model. Can we use signal driven IO as asynchronous IO? If yes, how? If no, why we cannot use signal IO as asynchronous IO?
20. Differentiate fcntl() and ioctl() system call. Explain the meaning of following socket

options: SO_BROADCAST,SO_KEEPALIVE,SO_RCVTIMEO, SO_SNDTIMEO, SO_REUSEADDR and SO_LINGER.

21. What are the socket options? Which functions are used to set and get a value of socket options? Explain them in detail with sample code.
22. What are the meanings of SIOCATMARK, SIOCGPGRP and SIOCSPGRP? How do you set socket as non-blocking using fcntl() and ioctl(). Illustrate with section of code.
23. What is daemon process and how does it started? How do you crate the daemon process in Unix? Explain.
24. What is syslogd()? What is the technique for logging messages from a daemon process? Explain with sample code.
25. How close() is different from shutdown()? Which functions set h_errono global variable? Differentiates errno and h_errno as well.
26. How windows socket is different from berkeley socket? Explain the role of setup(), cleanup() and WSAStartup() function in winsock architecture.
27. Explain winsock architecture. What are the types of dynamic linking in winsock programming?
28. Differentiate blocked I/O with unblocked I/O. Explain all the functions in detail to handle blocked I/O in winsock architecture.
29. What is overlapped I/O? Which function is used to set windows socket in non- blocking mode? Explain non-blocking socket with connect() in the case of winsock architecture.
30. Why WSAGetLastError() is required in winsock programming? Explain WSAGetOverlappedResult() function.
31. In which situations asynchronous IO is preferred than synchronous IO? Explain different Asynchronous IO functions in winsock programming.
32. How do you implement stream communication in Winsock? Describe each step with the help of relevant APIs.
33. Explain with the help of pseudo-code the use of accept with select such that the accept function doesn't block.
34. Write short notes on: telnet, tftp, ping, ifconfig/ipconfig, netstat, rlogin.

*** ALL THE BEST ***