Computer Networks, Assignment 2
Name: Samuel E George
Roll no. IIT2022007

Q2)

1)socket

- Synopsis: int socket(int domain, int type, int protocol);
- Parameters:
  - domain: Specifies the communication domain (e.g., AF_INET for IPv4, AF_INET6 for IPv6).
  - type: Specifies the socket type (e.g., SOCK_STREAM for TCP, SOCK_DGRAM for UDP).
  - protocol: Specifies the specific protocol to be used (usually 0 for default).
- Summary: Creates a new socket.

2)bind

- Synopsis: int bind(int sockfd, const struct sockaddr *addr, socklen_t addrlen);
- Parameters:
  - sockfd: Socket file descriptor obtained from socket().
  - addr: Pointer to a struct sockaddr containing the local address information.
  - addrlen: Size of the addr structure.
- Summary: Associates a socket with a local address.

3)connect

- Synopsis: int connect(int sockfd, const struct sockaddr *addr, socklen_t addrlen);
- Parameters:
  - sockfd: Socket file descriptor obtained from socket().
  - addr: Pointer to a struct sockaddr containing the remote address information.
  - addrlen: Size of the addr structure.
- Summary: Initiates a connection to a remote socket.

4)listen

- Synopsis: int listen(int sockfd, int backlog);
- Parameters:
  - sockfd: Socket file descriptor obtained from socket().
  - backlog: Maximum length of the queue of pending connections.
- Summary: Marks the socket referred to by sockfd as a passive socket, used to accept incoming connection requests.

5)accept

- Synopsis: int accept(int sockfd, struct sockaddr *addr, socklen_t *addrlen);
- Parameters:
    - sockfd: Socket file descriptor obtained from socket().
    - addr: Pointer to a struct sockaddr that will be filled with the address of the connecting entity.
    - addrlen: Initially contains the size of addr and is modified to reflect the actual size.
- Summary: Accepts a new incoming connection on a listening socket.

6)send

- Synopsis: ssize_t send(int sockfd, const void *buf, size_t len, int flags);
- Parameters:
    - sockfd: Socket file descriptor.
    - buf: Pointer to the data to be sent.
    - len: Length of the data in bytes.
    - flags: Additional options (usually 0).
- Summary: Sends data on a connected socket.

7)recv

- Synopsis: ssize_t recv(int sockfd, void *buf, size_t len, int flags);
- Parameters:
    - sockfd: Socket file descriptor.
    - buf: Pointer to the buffer where the received data will be stored.
    - len: Maximum length of the buffer.
    - flags: Additional options (usually 0).
- Summary: Receives data from a connected socket.

8)sendto

- Synopsis: ssize_t sendto(int sockfd, const void *buf, size_t len, int flags, const struct sockaddr *dest_addr, socklen_t addrlen);
- Parameters:
    - sockfd: Socket file descriptor.
    - buf: Pointer to the data to be sent.
    - len: Length of the data in bytes.
    - flags: Additional options (usually 0).
    - dest_addr: Pointer to a struct sockaddr containing the destination address.
    - addrlen: Size of the dest_addr structure.
- Summary: Sends data to a specific destination.

9)recvfrom

- Synopsis: ssize_t recvfrom(int sockfd, void *buf, size_t len, int flags, struct sockaddr *src_addr, socklen_t *addrlen);
- Parameters:
  - sockfd: Socket file descriptor.
  - buf: Pointer to the buffer where the received data will be stored.
  - len: Maximum length of the buffer.
  - flags: Additional options (usually 0).
  - src_addr: Pointer to a struct sockaddr that will be filled with the source address.
  - addrlen: Initially contains the size of src_addr and is modified to reflect the actual size.
- Summary: Receives data from a specific source.

10)close

- Synopsis: int close(int fd);
- Parameters:
  - fd: File descriptor to be closed.
- Summary: Closes a file descriptor, including sockets.

11)shutdown

- Synopsis: int shutdown(int sockfd, int how);
- Parameters:
  - sockfd: Socket file descriptor.
  - how: Specifies the type of shutdown (e.g., SHUT_RD for reading, SHUT_WR for writing, SHUT_RDWR for both).
- Summary: Shuts down part or all of a full-duplex connection associated with the socket.

12)fork

- Synopsis: pid_t fork(void);
- Parameters: None.
- Summary: Creates a new process (child) by duplicating the calling process (parent). The child process gets a copy of the parent's address space.