Вот краткие ответы на вопросы по лабораторной работе на тему "Программирование сетевых приложений".

1. \*\*Сетевые утилиты\*\*: `ping`, `traceroute`, `netstat`, `ipconfig`/`ifconfig`, `nslookup`, `dig`, `telnet`, `ftp`, `ssh`, `scp`, `netcat (nc)`, `curl`, `wget`.

2. \*\*Уровни модели OSI/ISO\*\*:

- \*\*Физический\*\* — передача битов по физической среде.

- \*\*Канальный\*\* — передача кадров между узлами, контроль ошибок.

- \*\*Сетевой\*\* — маршрутизация данных между сетями.

- \*\*Транспортный\*\* — обеспечение надежной доставки данных.

- \*\*Сеансовый\*\* — управление сессиями, установление и завершение соединений.

- \*\*Представительный\*\* — преобразование данных (например, шифрование).

- \*\*Прикладной\*\* — взаимодействие с приложениями (например, HTTP, FTP).

3. \*\*Сетевой протокол\*\* — это набор правил и стандартов, определяющих способ обмена данными между устройствами в сети.

4. \*\*Граница между аппаратным и программным обеспечением\*\* в модели OSI/ISO проходит между канальным и сетевым уровнями.

5. \*\*CSMA/CD (Carrier Sense Multiple Access with Collision Detection)\*\* — протокол, используемый в Ethernet для обнаружения коллизий при передаче данных.

6. \*\*Программное обеспечение для подуровня LLC\*\* — это драйверы сетевого интерфейса, которые реализуют логическое управление каналом (Logical Link Control).

7. \*\*NDIS (Network Driver Interface Specification)\*\* — спецификация, определяющая интерфейс между сетевыми драйверами и операционной системой.

8. \*\*Свойства ненадежных протоколов\*\*: отсутствие гарантии доставки, нет подтверждения о получении данных. Примеры: ненадежные — UDP, надежные — TCP.

9. \*\*Уровни TCP/IP\*\*:

- \*\*Прикладной\*\* — протоколы приложений (HTTP, FTP, DNS).

- \*\*Транспортный\*\* — доставка данных (TCP, UDP).

- \*\*Сетевой\*\* — маршрутизация (IP).

- \*\*Канальный\*\* — физическая передача данных.

10. \*\*Хост\*\* — устройство, подключенное к сети. \*\*Адрес хоста\*\* — IP-адрес устройства. \*\*Имя хоста\*\* — доменное имя, которое может использоваться вместо IP-адреса.

11. \*\*Поддержка протоколов Internet\*\* — IETF. Документы, описывающие протоколы, называются RFC (Request for Comments).

12. \*\*MAC-адрес\*\* — уникальный физический адрес сетевого устройства. Структура: 48 бит (6 байт), разделена на 2 части: OUI (идентификатор производителя) и NIC (идентификатор устройства).

13. \*\*Просмотр MAC-адреса\*\*: `ipconfig /all` (Windows), `ifconfig` (Linux/Mac).

14. \*\*Межсетевой уровень\*\* — отвечает за маршрутизацию пакетов между сетями (уровень IP).

15. \*\*Структура IP-адреса\*\* — делится на две части: сеть и хост. IPv4 — 32-битный адрес, IPv6 — 128-битный.

16. \*\*Типы IP-адресации\*\*: публичная и частная. \*\*Классы адресов Internet\*\*: A, B, C, D, E.

17. \*\*Публичный IP-адрес\*\* — уникален в Интернете, \*\*частный IP-адрес\*\* — используется внутри локальных сетей.

18. \*\*Просмотр IP-адреса\*\*: `ipconfig` (Windows), `ifconfig` (Linux/Mac).

19. \*\*Тестирование IP-соединения\*\*: с помощью команды `ping`.

20. \*\*Список сетевых узлов между двумя хостами\*\*: команда `traceroute` (или `tracert` в Windows).

21. \*\*Параметры настройки TCP/IP\*\*: IP-адрес, маска подсети, шлюз по умолчанию, DNS-сервер.

22. \*\*Маска подсети\*\* — определяет, какая часть IP-адреса относится к сети, а какая — к хосту.

23. \*\*Основные отличия IPv4 и IPv6\*\*: длина адреса (32 бита vs 128 бит), отсутствие NAT в IPv6, встроенная поддержка шифрования и QoS в IPv6.

24. \*\*Сетевой порт\*\* — это число, идентифицирующее конкретное приложение на устройстве. Понятие определено на транспортном уровне модели TCP/IP.

25. \*\*Классификация сетевых портов\*\*: известные порты (0-1023), зарегистрированные порты (1024-49151), динамические/частные порты (49152-65535).

26. \*\*Просмотр программ, использующих порты\*\*: `netstat -an` или `lsof -i`.

27. \*\*Архитектура клиент/сервер\*\* — модель взаимодействия, где клиент запрашивает услуги у сервера.

28. \*\*Сетевая служба\*\* — программа, предоставляющая функции по сети (например, DNS, FTP, HTTP).

29. \*\*Интерфейс внутренней петли (loopback)\*\* — это интерфейс, через который система может отправлять пакеты самой себе (адрес 127.0.0.1).

30. \*\*Назначение DNS и DHCP\*\*:

- \*\*DNS\*\* — преобразование доменных имен в IP-адреса.

- \*\*DHCP\*\* — автоматическая выдача IP-адресов устройствам в сети.

31. \*\*Организация IP-адресов\*\* — IANA. Домены верхнего уровня поддерживает ICANN.

32. \*\*Сетевой сокет\*\* — программный интерфейс для передачи данных между узлами сети.

33. \*\*POSIX (Portable Operating System Interface)\*\* — стандарт для обеспечения совместимости между различными ОС.

34. \*\*Структура TCP-сервера\*\*:

- `socket()`, `bind()`, `listen()`, `accept()`, `recv()`, `send()`, `close()`.

35. \*\*Структура TCP-клиента\*\*:

- `socket()`, `connect()`, `send()`, `recv()`, `close()`.