

P1 ans:

Bernie的速度为:

$$v = 18km/h = 5m/s = 0.005km/s$$

Bernie携带的磁带总容量（单位由字节转成位）为:

$$D = 3 * 7GB * 8 = 168GB$$

假设在xkm内Bernie的数据传输速率会超过数据线，则移动这段距离所需要的时间为:

$$T = \frac{x}{v} = \frac{x}{0.005} = 200x \text{ s}$$

Bernie的数据传输速率就等于:

$$D_V = \frac{D}{T} = \frac{168GB}{200x \text{ s}}$$

根据题目要求可知，Bernie的数据传输速率要大于150Mbps，所以:

$$D_V > 150Mbps$$

$$\frac{168GB}{200xs} > 150Mbps$$

即（转换成相同单位——Mbps）:

$$\frac{840}{x} > 150$$

解得:

$$x < 5.6$$

所以在距离小于5.6km的范围内，Bernie的速率会大于传输线

P2 ans:

因为无线电是一种电磁波，所以其速度近似认为是:

$$v = 3 * 10^8 \text{ m/s}$$

卫星高度等于:

$$H = 40000 \text{ km} = 4 * 10^7 \text{ m}$$

因为客户机发出请求和服务器做出响应都需要使用卫星网络，所以

$$T = \frac{4 * H}{v} = \frac{4 * 4 * 10^7 \text{ m}}{3 * 10^8 \text{ m/s}} = 0.533 \text{ s}$$

所以最佳情况下的延迟是0.533s，也就是533ms。

### P3 ans:

#### 使用层次协议的两个理由:

- 各层之间是独立的: 某一层并不需要知道它的下一层是如何实现的, 而仅仅需要知道该层通过层间的接口(即界面)所提供的服务。由于每一层只实现一种相对独立的功能, 因而可将一个难以处理的复杂问题分解为若干个较容易处理的更小一些的问题, 使问题的复杂性降低。
- 灵活性好: 当任何一层发生变化时, 只要层间接口关系保持不变, 则在这层以上或以下层均不受影响。当某一层出现技术革新或者某一层在工作中出现问题时不会连累到其它层的工作, 排除问题时也只需要考虑这一层单独的问题即可。

#### 层次协议可能的缺点

- 性能开销: 每一层都会添加一些额外的处理步骤和数据封装, 这可能导致整体性能的降低。例如, 数据包在通过网络时需要在每一层进行封装和解封, 这增加了处理时间和潜在的延迟

### P4 ans:

- TCP (传输控制协议) 是一个**可靠的、面向连接**的协议, 确保从一台机器发出的字节流能够**正确无误**地交付到互联网上的另一台机器。它将注入的字节流**分割成离散的报文**, 每个报文都传递给互联网层。此外, TCP还负责**处理流量控制**, 防止快速发送方因发送过多报文而淹没处理能力不足的慢速接收方。每个TCP连接都**有一个数据包大小限制** (MSS, 最大报文段大小), 使得数据传输过程较为复杂。因此, 它适用于**对数据传输可靠性和准确性要求较高的应用**, 如网页浏览、文件传输 (FTP)、电子邮件 (SMTP)、数据库同步等。
- UDP (用户数据报协议) 则是一个**不可靠的、无连接**的协议, **不保证数据包的顺序或完整性**, 仅将数据包发送到目的地, 而不考虑是否成功到达。UDP协议适用于那些对可靠性要求不高, 但对**传播速度和延迟敏感**的应用, 如视频会议、直播、游戏等。尽管**支持更大的数据包**, 每个UDP数据报的大小也通常应在网络最大传输单元 (MTU) 范围内。

### P5 ans:

- **方案一:**
  - **\*\*优点:\*\***提高传输效率, 因为每个数据包都可以立即得到确认, 避免了等待整个文件传输完成的时间, 如果某个数据包丢失或损坏, 只需重新发送该数据包, 而不需要重传整个文件, 节省了时间和带宽。
  - **\*\*缺点:\*\***用户没办法验证接受到的文件的完整性和正确性, 如果某个数据包丢失或错误, 需要重新传输, 这可能会增加总的传输时间。而且每个数据包都需要独立的确认, 无疑增加了网络负载和处理时间。
- **方案二:**
  - **\*\*优点:\*\***只需要在文件传输完成后进行一次确认, 减少了网络负载和处理时间, 由于整个文件作为单一实体处理, 一旦确认就意味着整个文件都正确无误地到达了目的地。
  - **\*\*缺点:\*\***在大型文件传输过程中, 如果发生错误或丢包, 发送方无法及时发现并纠正, 可能导致长时间的等待和低效的重传, 如果文件很大且传输过程中出现问题, 可能需要重新传输整个文件, 而不是仅重传有问题的部分。

这两种策略各有优缺点, 具体使用哪种策略取决于具体的应用场景和需求。例如, 对于需要高实时性和快速反馈的场景, 可能更倾向于使用第一种策略; 而对于大文件传输或对完整性要求极高的场景, 可能更倾向于使用

第二种策略。

P6 ans:

原始长度为M，经过第一层以后就会变成M+h，经过第二层以后就会变成M+2h...依此类推，第n层以后就会变成M+nh，所以报文头所占的网络带宽比例就是：

$$\frac{n * h}{M + n * h}$$