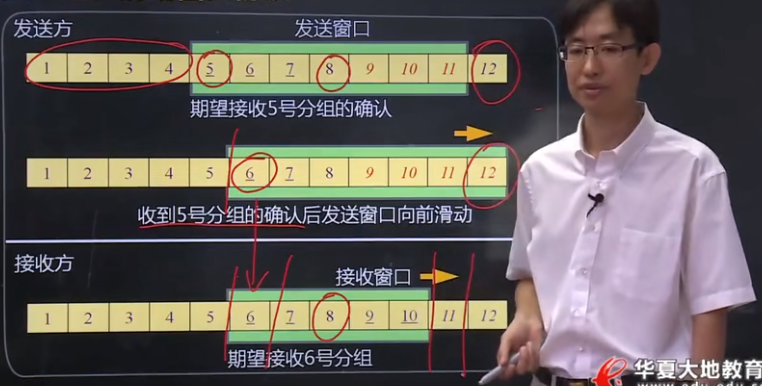
停-等协议与滑动窗口协议（如何保证通信的可靠性）

因为不是传输层自己进行信息的传输，而是由网络层进行信息的传输，就像是自己不发快递而是由快递公司进行发送，解决怎么发现数据丢失，怎么解决丢失问题等问题

1. 可靠数据传输基本原理
2. 差错检测：利用差错编码实现数据包传输过程中的比特差错检测
3. 确认：接收方向发送方反馈接收状态
4. 重传：发送方重新发送接收方没有正确接收的数据
5. 序号：确保数据按序提交（保证数据不重复）
6. 计时器：解决数据丢失问题（多长时间接收方还没有发送收到的信息就自动认为丢失）
7. 停-等协议

停-等协议的主要特点就是每发送一个报文段后就停下来等待接收方的确认

1. 发送方发送经过差错编码和编号的报文段，等待接收方的确认（发送并等待确认）
2. 接收方如果正确接收报文段，即差错检测无误且序号正确，则接受报文段，并向发送方发送ACK，否则丢弃报文段，并向发送方发送NAK（接收并确认/否认）
3. 发送方如果收到ACK，则继续发送后续报文段，否则重发刚刚发送的报文段（继续发送/重发）
4. 滑动窗口协议
5. 停-等协议的主要性能问题：停止-等待机制降低了信道利用率
6. 解决方法：流水线协议或管道协议——允许发送方在没有收到确认前连续发送多个分组
7. 流水线协议的改进：增加分组序号范围，发送方和（或）接收方必须缓存多个分组（防止没有发送成功（或接收成功）就需要从本地重新发送（或接收））
8. 典型的流水线协议：滑动窗口协议
9. 滑动窗口协议



两种最具有代表性的滑动窗口协议：

1. 回退N步（Go-Back-N，GBN）协议：发送端窗口大小较大，可以在未得到前连续发送多个分组；但接收窗口大小仅为1，只能接收1个按序到达的分组，未按序到达的分组或者某个分组差错，就会引起发送方重发该分组及其之后的所有分组
2. 选择重传（Sekective Repeat，SR）协议：增加接收方缓存能力（接收窗口>1），缓存正确到达但失序的分组，仅要求发送方重传未被接收方确认的分组，等缺失分组到达后一并向上层按序提交

