

【分组要求:】

- 1、以小组为基本单位，每个小组人数上限为 3 人，自由组合
- 2、以 G00103 作业的小组组合为基准，如无变动则**不再申报**，如有变动可**再次申报**
- 3、考虑到不同人数的工作量差别，如果某个作业的分值为 n ，则三人小组，每人得分为 $n \times \text{得分率}$ ；两人小组，每人得分为 $n \times \text{得分率} \times 1.1$ ；单人小组，每人得分为 $n \times \text{得分率} \times 1.2$
- 4、每个小组成员的得分相同，不会因为贡献大小而区别给分
- 5、再次申报小组成员的截止时间为 **12 月 2 日 23:59:59**，如果截止时间后未申报，则维持不变

【作业提交要求:】

- 1、每个小组仅需要提交一次，由学号最小的同学负责提交即可，其余同学不需要提交
- 2、如果该组学号最小的同学未提交而有其他人提交，则**无效**，视为未提交
- 3、提交文件的压缩包子目录展开后**必须**是提交人的学号

【数据传递过程描述如下: (本部分仅为参考建议, 各组也可自行设计)】

- 1、网络层、数据链路层、物理层用不同的进程进行模拟，相互之间的数据传递用文件进行模拟（与个人作业不同）
- 2、后续文档描述中各层简写约定如下：
 - a) 发送方网络层: SNL
 - b) 发送方数据链路层: SDL
 - c) 发送方物理层: SPL
 - d) 接收方物理层: RPL
 - e) 接收方数据链路层: RDL
 - f) 接收方网络层: RNL
- 3、各层之间的事件通知用信号完成
 - a) timeout 用 SIGALRM 信号，精度在 ms 级（不要用 alarm 函数）
 - b) ack_timeout 也用 SIGALRM 信号（如何与 timeout 区分？）
 - c) chsum_err 用自定义的 35 号信号（kill -l 显示 SIGRTMIN+1）
 - d) frame_arrival 用自定义的 36 号信号
 - e) network_layer_ready 用自定义的 37 号信号
 - f) enable_network_layer() 由 SDL 向 SNL 发送 38 号信号
 - g) disable_network_layer() 由 SDL 向 SNL 发送 39 号信号
- 4、SNL 与 SDL 之间的数据交互要求
 - a) SNL 的数据来源为某个指定的文件（可以 1GB 以上）
 - b) SNL 为了保证向下层 SDL 有足够的发送数据，将数据每 1024 个字节写为一个共享文件，文件名从 network_datalink.share.0001~network_datalink.share.0999（如果传输速度足够快，可扩展至 9999），由 SNL 按顺序轮询写，SDL 按顺序顺序读
 - c) SNL 初始为 enable 状态，如果收到 39 号信号，则切换为 disable 状态
 - d) SNL 如果在 enable 状态，则依次取指定文件中的 1024 字节写成一个共享文件（文件名: network_datalink.share），向 SDL 发送 network_layer_ready 信号（前四个算法忽略此信号，后两个算法要处理），随后进入阻塞（注意方法，不要占死 CPU 资源），如果在 disable 状态，则对文件加独占锁，直到 enable 状态才释放
 - e) SDL 如果在 SNL 处于 disable 状态时调用 from_network_layer()，则直接锁死（程序挂起，本次测试失败）
 - f) SDL 如果在 SNL 处于 enable 状态时调用 from_network_layer()，则首先读共享文件加独占锁（network_datalink.share），读取 1024 字节后，释放独占锁，并用 38 号信号通知 SNL
 - g) SNL 收到 38 号信号后，跳转到步骤 b
 - h) 如果最后一次的数据不满 1024，如何处理？请自行设计

5、SDL 与 SPL 之间的数据交互要求

(略, 请各组自行补充完成)

6、SPL 和 RPL 之间的数据交互要求 (为达到不同组程序互测的目的, 此项必须一样, 不准改变)

a) 用理想状态的网络环境中基于 TCP 的数据传输来模拟物理层的传输

b) 包长分为两种

i. 含数据的包, 长度为 1036 字节 (依次为帧类型 4+发送序号 4+接收序号 4+数据 1024)

ii. 不含数据的包, 长度为 12 字节 (依次为帧类型 4+发送序号 4+接收序号 4)

iii. 为了便于区分, 约定纯 ACK/NAK 包, 发送序号为 0xFFFFFFFF

c) TCP 写采用阻塞方式, 每次 12/1036 字节

d) TCP 读采用阻塞方式, 每次 12/1036 字节

e) 最后一个不满 1024 的数据包, 填充 '\0' 至数据满 1024 为止 (如果文件恰好是 1024 的整数倍, 则不需要)

f) 发完最后一个包后, 再发一个 1024 字节的全 '\0' 数据包, 表示传输结束 (假设正文不含 1024 全\0)

g) 人为保证传送文件不是 '\0' 结尾, 即倒数第二个包最后的 '\0' 去除即可

7、RPL 和 RDL 之间的数据交互要求

(略, 请各组自行补充完成)

8、RDL 和 RNL 之间的数据交互要求

(略, 请各组自行补充完成)

【题目要求:】将第三章最后的 6 个算法用程序模拟实现出

1、每个小组作业的目录结构要求如下 (假设小组学号最小为 1651234, 各组按实修改): 首先建立 "1651234-G00104" 目录, 可位于任意子目录下, 下面再建立 7 个子目录, 示例如下:

```
1651234-000110
|-- 01
|-- ...
|-- 06
|-- common
```

2、公共函数目录 (common)

- 存放各协议用到的公共函数
- 不单独生成可执行文件

3、协议 1: 乌托邦式的单工协议 (01 子目录)

- 编写若干程序及 makefile 文件, 要求 make 后生成 sender1_network、sender1_datalink、send1_physical、receiver1_network、receiver1_datalink、receiver1_physical 共六个文件, 分别模拟发送方和接收方的三层协议 (物理层由 Send1 向 receiver1 发起连接)
- 如果采用 fork 进程方式 (例如: make 后生成 sender1 可执行文件, 执行后 fork 出 send1_network、send1_datalink、send1_physical 三个进程; make 后生成 receiver 可执行文件, 执行后 fork 出 receiver1_network、receiver1_datalink、receiver1_physical 三个进程, 则 make 后只需要 sender1 和 receiver1 两个可执行文件即可), 则可执行文件数量可以少, 但是执行时一定要收发共六个进程!!! (下同)

4、协议 2: 无错信道上的单工停-等协议 (02 子目录)

- 编写若干程序及 makefile 文件, 要求 make 后生成 sender2_network、sender2_datalink、send1_physical、receiver2_network、receiver2_datalink、receiver2_physical 共六个文件, 分别模拟发送方和接收方的三层协议 (物理层由 Send2 向 receiver2 发起连接)
- receiver2 要有一定概率 (10%) 延时发 ACK (模拟接收方处理能力有限)

5、协议 3：有错信道上的单工停-等协议（03 子目录）

- 编写若干程序及 makefile 文件，要求 make 后生成 sender3_network、sender3_datalink、send3_physical、receiver3_network、receiver3_datalink、receiver3_physical 共六个文件，分别模拟发送方和接收方的三层协议（物理层由 Send3 向 receiver3 发起连接）
- RPL 要求有一定概率（3%）收到 SPL 的正确数据后丢弃，向上发送 cksum_err
- RPL 要求有一定概率（3%）收到 SPL 的正确数据后丢弃，不向上发送（模拟数据丢失）
- SPL 要求有一定概率（3%）收到 RPL 的 ACK 包后丢弃，向上发送 cksum_err
- SPL 要求有一定概率（3%）收到 RPL 的 ACK 包后丢弃（模拟 ACK 丢失）

6、协议 4：1 位窗口滑动协议（04 子目录）

- 编写若干程序及 makefile 文件，要求 make 后生成 protocol4_network、protocol4_datalink、protocol4_physical_c、protocol4_physical_s 共四个执行文件（由 c 向 s 发起连接）
- RPL 和 SPL 的模拟错误要求同协议 3

7、回退 N 协议（05 子目录）

- 编写若干程序及 makefile 文件，要求 make 后生成 protocol5_network、protocol5_datalink、protocol5_physical_c、protocol5_physical_s 共四个执行文件（由 c 向 s 发起连接）
- RPL 和 SPL 的模拟错误要求同协议 3

8、选择重传协议（06 子目录）

- 编写若干程序及 makefile 文件，要求 make 后生成 protocol6_network、protocol6_datalink、protocol6_physical_c、protocol6_physical_s 共四个执行文件（由 c 向 s 发起连接）
- RPL 和 SPL 的模拟错误要求同协议 3，其中丢弃的 ACK 要引发 ack_timeout

【本次作业的统一批改方法说明：】

1、每个人的目录结构要求如下（假设学号为 1651234，各人按实修改）：首先建立“1651234-G00104”子目录，下面再有 01-06、common 共 7 个子目录

2、提交作业时，每位同学上交一个 linux-data-link-layer.tar.bz2 文件，解压后能得到上述的完整目录结构，截止时间到后，会从每人的交作业目录中复制出来，全部放在 16-G00104 目录中
示例如下：

```
16-G00104
|-- 1651234-linux-data-link-layer.tar.bz2      （第 1 组的作业压缩包）
...
`-- 1654321-linux-data-link-layer.tar.bz2      （最后 1 位同学的作业压缩包）
```

依次解压后，能得到如下目录结构：

```
16-G00104
|-- 1651234-G00104                            （第 1 组同学的作业目录）
...
`-- 1654321-G00104                            （最后 1 组同学的作业目录）
```

3、进入 16-G00104 目录，进行一次 check.sh，就能生成所有可执行文件，示例如下：

```
16-G00104
|-- 1651234-G00104                            （第 1 组同学的作业目录）
...
|-- 1654321-G00104                            （最后 1 组同学的作业目录）
`-- check.sh                                  （老师事先建好的 shell 文件，准备编译所有组的本次作业，具体的实现方式是进入到每个学号对应的目录后调用该目录下的总 makefile）
```

4、无法顺利编译则不能得分，对应学号及子目录名错则不能得分

5、作业提交时清除所有的中间文件及生成的可执行文件、源程序备份文件等

【检查通过标准（要求不同组互测）：】

- 1、所有协议均要求写日志，清除记载各种错误及事件的发生，包括第几个包错误，重发了几个，接收端如何排除重复等
- 2、指定一个 1GB 以上的文件（二进制格式），传输后双方文件比较一致即为通过

【实验报告撰写：】

- 1、本次作业需要撰写实验报告，具体内容包括：自行设计的规则部分、完成过程中碰见的问题
- 2、实验报告单独提交

【作业要求：】

- 1、**12月9日前**网上提交(考虑到测试，截止时间为周日)
- 3、每题所占平时成绩的具体分值见网页（小组作业单独计分，总分为 100）
- 4、超过截止时间提交作业则不得分