Introduction to Computer Network

Lab2

110030018 林芷儀

**實作過程:**

**Server:**

從助教提供的模板和hint逐漸加入如下功能:

1. ACK timer和Timeout retransmission:
   * 用clock\_t 的變數 t\_begin, duration來記錄初始時間和經過時間
   * 每次while(1)的迴圈中都檢查一次timeout
   * 如果timeout就重傳
   * 如果沒有timeout，確認「成功送出封包」並確認「收到正確的ack number的封包」，都正確的話就繼續送下一份封包
2. Seq number和影片資料讀取
   * 每次讀取1024bute大小的資料，放入封包的data區
   * 用變數count / remain\_len計算當前剩餘的資料大小
   * 如果讀到最後一個封包，只讀入剛好大小的data到封包中，送完該封包後就跳出迴圈，表示傳送完成
3. 修改udp packet的isLast型態為 unsigned int。
   * 主要目的是為了讓server sequence為0的封包，把整個video file的檔案大小資訊傳給client，方便client端計算剩餘的封包大小。
   * 雖然isLast本身不是用來這麼做的，但為了盡量不改動原本的架構，我覺得這是比較好的方式。

**Client:**

從助教提供的模板和hint逐漸加入如下功能:

1. 修改udp packet的isLast型態為 unsigned int，理由同上
2. 用變數rcv確認收到的封包是正常的
3. 收到之後，用isLoss來模擬packet loss
4. 如果沒有封包遺失，用變數receive\_packet來確認收到的是正確的seq number的封包。如果正確就寫入file中，並回傳Ack給server端。
5. 封包傳送過程中不斷更新變數getFilesize，並依此決定讀取資料的大小。
6. 若從server端拿來的封包發現是最後一個(is\_last flag==1)，只讀入剛好大小的資料(藉由getFilesize變數)，並跳出迴圈，代表完成傳輸。

**編譯及執行方式:**

用makefile: make

**實作結果:**



