

Implementation

9817024 陳上奎

整個傳檔案的過程(Stop-and-wait)可以分成兩個部分(A,B)再分別三種不同的 timeout(A,B,C)來實作：

A. Client

因為傳送時是一個一個 packet 而 packet 的大小又有限制，因此必須將檔案分割成許多個片段。

1. 先將要傳送的 packet 準備好，裡面必須要有第幾個片段、是否是檔案的結尾、所攜帶的資料大小和攜帶的資料，如果發現檔案結束又收到連續 10 個 timeout 則視為傳送完成
2. 傳送 packet
3. 接收回傳的 packet，裡面會有接收到第幾個的片段，如果值和應該收的值不同則重傳現在 packet
4. 當接收發生 timeout 時則重傳現在的 packet

B. Server

1. 接收由 client 傳過來的 packet，如果傳過來的片段是正確的則寫入檔案中，並回傳這個片段的值，如果有錯則重傳上一個正確的片段的值
2. 當接收發生 timeout 時則重傳上一個正確的片段的值
3. 如果接收的 packet 是檔案結尾則視為傳送完成

A. Timeout using SIGALRM

利用 `alarm()` 來觸發 SIGALRM，並利用 signal 來 implementation 當

SIGALRM 觸發時的處理情況，可是在 fedora 內並不會觸發 interrupt，

因此需補上 `siginterrupt()` 來處理 SIGALRM 的 interrupt

B. Timeout using select

利用 `select()` 來看在 timeout 時間內有無 packet

C. Timeout using setsockopt

利用 `setsockopt()` 讓 `recvfrom()` 在 timeout 後自動中斷