111060005 胡昱煊 lab3

Description:

The client.c and server.c file included are the required files to this lab. They can be built by the “Make” tool and do meet all the requirements the lab wants.

There are three functions in the two programs that we need to work on: “recvFile” and “writeFile” from client.c, and a “sendFile” from server.c .

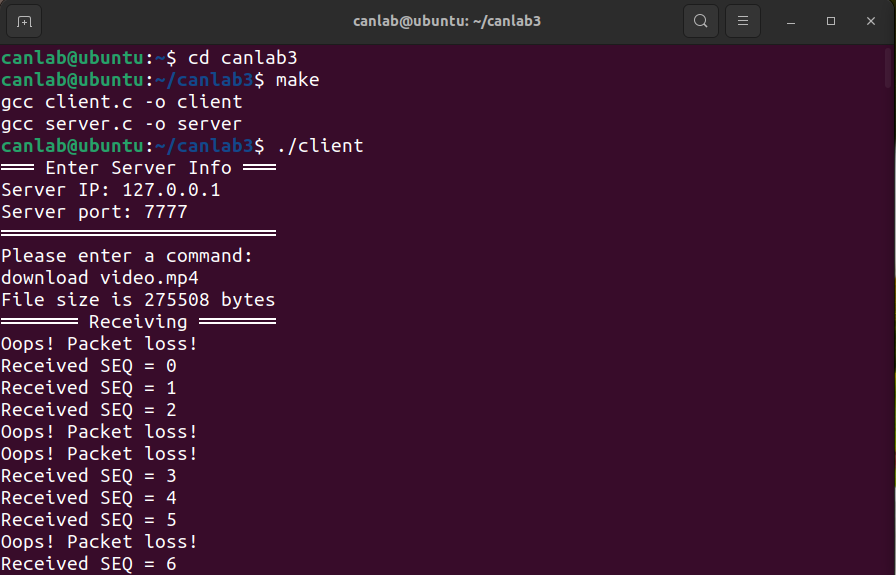
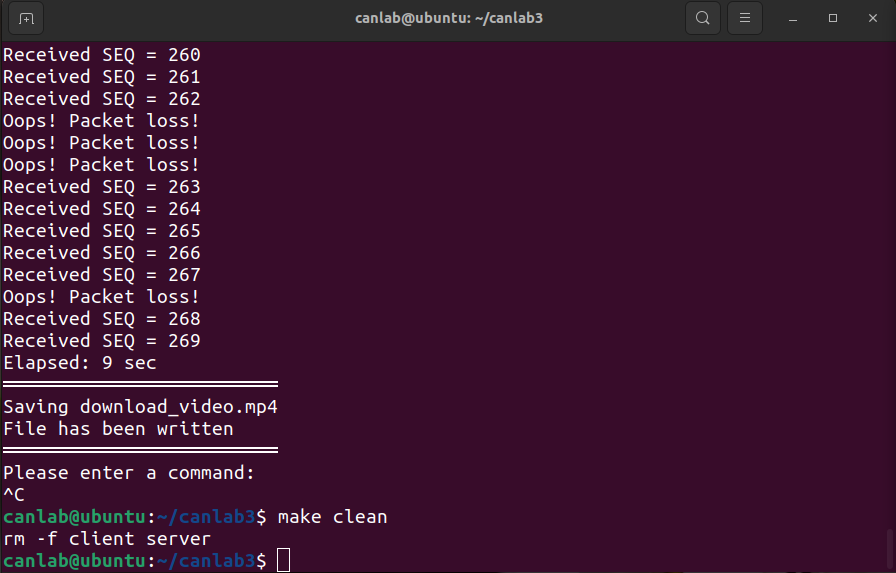
Generally, I wrote all the three functions based on the instructions the templates provide. At the meantime, since the functions require read and write data from/to a file, I had to look up the internet about file-related functions such as fwrite(), fseek(), etc.

What’s worth mentioning is that when I first complete the three functions, the packets were successfully sent, but the bytes in the “download\_video.mp4” are totally wrong. After some checks (by sending some self-written .txt files), I found out that the “buffer” in “recvFile” function only recorded the content of the last packet sent. Therefore, I asked chatgpt about the solution and then solved the problem.

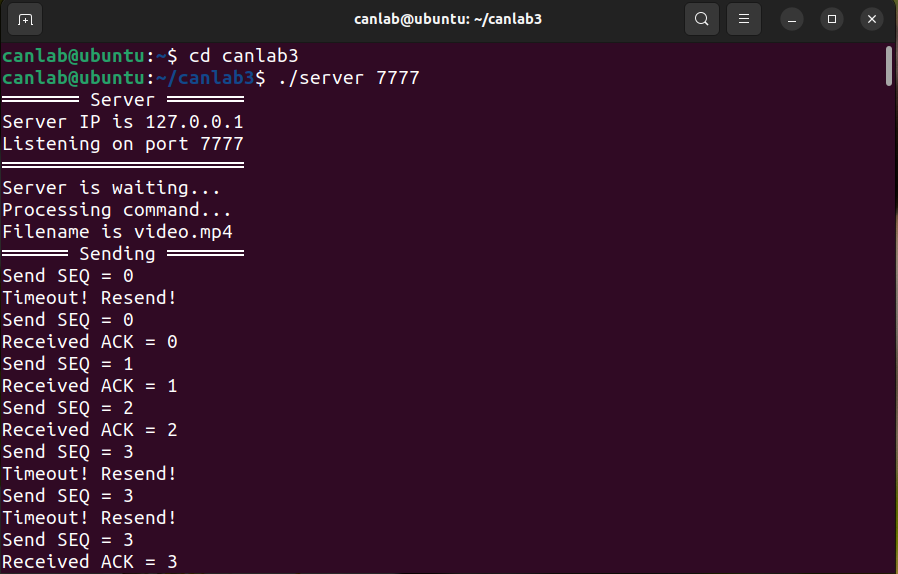
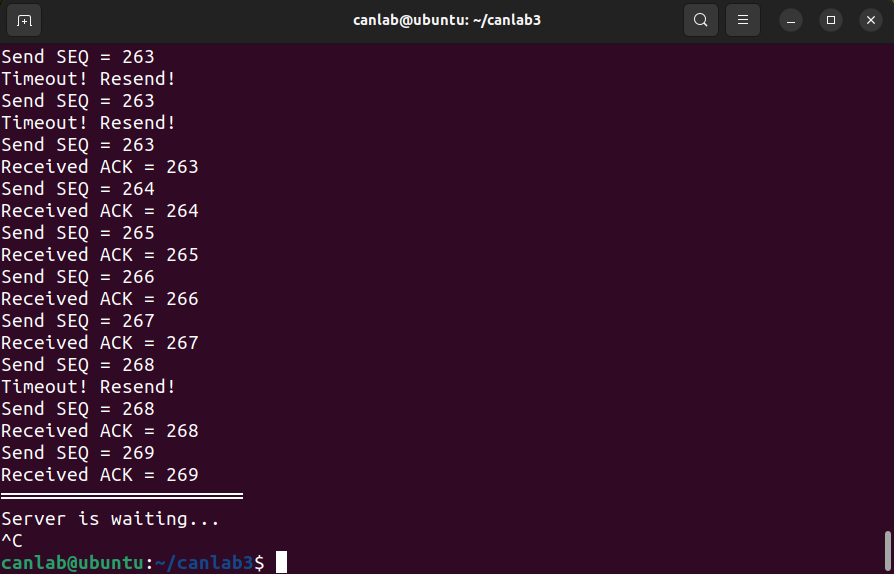
In this lab, I’ve learned how to implement stop-and-wait mechanism with C socket programming.

The picture below are the screenshots required:

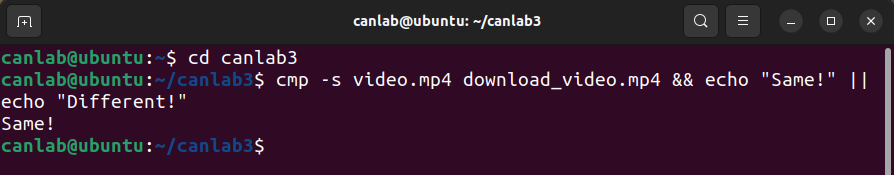
1. Client side and make file executability



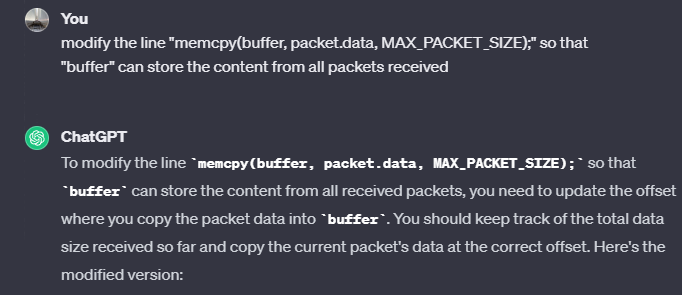
2. Server side



3. Comparison between original and downloaded file



4. Chatgpt usage in this lab



Reference:

(1) C速查手冊

<http://kaiching.org/pydoing/c/c-std-string.html>

(2) fwrite 和 fread函数的用法小结

<https://www.runoob.com/w3cnote/c-programming-fwrite-and-fread-summary.html>

(3) Lab3.pdf

(4) UDP传输 ：recvfrom 函数与 sendto 函数分析

<https://zhuanlan.zhihu.com/p/408369874>

(5) poll <https://book.itheima.net/course/223/1277519158031949826/1277529226395787267>

(6) C 库函数 - fseek()

<https://www.runoob.com/cprogramming/c-function-fseek.html>

(7) fopen() - C語言庫函數

<https://tw.gitbook.net/c_standard_library/c_function_fopen.html>

(8) C 库函数 - fclose()

<https://www.runoob.com/cprogramming/c-function-fclose.html>