

Solution

A1

It's possible to build a chat program with 2 threads. One thread is use to record the input and sent message. Another is use to recieve message form the opposite chat program. (Since waiting for input and reading socket will both block the thread, so, these two operations can not happen at the same time and thus we need 2 threads for them).

We can descibe the chat program in the following way. First Host1 listen on port [Port1] with [Thread11]. Host2 connect to Host1 on [Port1] with [Thread21]. Then Host2 creates a new thread named [Thread22] and starts to listen on port [Port2]. Host1 creates a new thread named [Thread12] and connect to Host2 on [Port2]. Host1 recieve messages with [Thread11] and send messages with [Thread12]. Host2 recive messages with [Thread22] and send messages with [Thread21].

A2

To transfer file with UDP, we need to ensure that every part of the file is sent and recieved. In practical, we divide a file into a sequence of packages and assign a number to each package and set the pakge together with the number to clients. When client recieve the packages, they check wheather there is a missing number. If so, they can ask the server for that part again. If not, the clients save the packages in the order of the package number.