CHAT SERVER

Design a chat server using TCP such that the screen of your computer is divided in two parts with borders. One part will have send the message and another part will have receive the message.

PROGRAMS : chat_serv.c, chat_client.c.
LOCATION : chat
HOW TO RUN : At first run server (chat_serv.c) in one terminal, and then run different clients (chat_client.c) on different terminal. Give server port number as an argument on the server's terminal and give server's port number and client's port number as two argument on the client's terminal.

INPUT / OUTPUT

1st CLIENT	SERVER
[chat]#cc chat_client.c -o c [root@infosysroot@infosys chat]#./ c 5000 6000 Send message to server : Hello i am Ranjeeta data send (Hello i am Ranjeeta) Received message from server : I am Kan. Send message to server : quit data send (quit) Closing connection with the server. [root@infosys chat]#_	[root@infosys chat]#cc chat_serv.c -o s [root@infosys concurrent]#./s 5000 Successfully created stream socket. Bound locak port successfully. Waiting for client connection on port TCP 5000 Received connection from host [IP 127.0.0.1, TCP port 6000] Waiting for client connection on port TCP 5000 Received from host [IP 127.0.0.1, TCP port 6000] : Hello i am Ranjeeta Enter string for client : I am Kan. Waiting for client connection on port TCP 5000 Received from host [IP 127.0.0.1, TCP port 6001] : How r u Enter string for client : quit Closing connection with host [IP 127.0.0.1, TCP port 6001] Received from host [IP 127.0.0.1, TCP port 6000] : quit Closing connection with host [IP 127.0.0.1, TCP port 6000]
2nd CLIENT	
[root@infosys chat]#cc chat_client.c -o c [root@infosys chat]#./ c 5000 6001 Send message to server : How r u data send (How r u) Closing connection with the server. [root@infosys chat]#_	