Process Messaging

This project requires that you write a C or C++ application to share console messages with various processes. The program should allow the user to type in a message and it should print out all messages received from other processes. This should work no matter how many programs you start.

1. Your program **shall** read messages from standard input.
   (a) Your program **shall** provide all messages entered by the user to other processes.
   (b) Your program **shall** provide the sender process ID to the other processes.
   (c) Your program **shall** terminate if the user types *exit*. Do not send this as a message. This is the only way your program terminates (besides help).
2. Your program **shall** print all information received from other processes. Do not print messages typed into your process. You should print the sender process ID and the string message.
3. Your program **shall** use at least one form of inter-process communication other than a file. I don't care what IPC facility you use, but you must use one to communicate between all of the spawned processes. If you use files, you must also use a different IPC facility for something legitimate (control, background processing, etc).
4. Your program **shall** execute at least two threads. You must spawn an additional thread besides your main thread.
5. Your program **shall** support unlimited connected processes. If the IPC facility you choose limits how many, that is okay. You should not only work with two connected processes. You should be able to support at least 5 regardless of the IPC facility you use.
6. Your program **shall** support printing help information. You must support the -h option. This option must print how to use your program and then terminate. Do not create any IPC or threads or anything else.
7. Your program **shall** clean up all resources it uses. This means that after I terminate the last program running, none of your IPC objects should be left around. Additionally, it should correctly close all files and clean up any threads or processes it creates.
8. Your program **shall** meet the basic requirements listed below:
   (a) Your program **shall** print reasonable error information.
   (b) Your program **shall** terminate after encountering an unrecoverable error (after printing error information).
   (c) You **shall** submit all source code and a makefile in a tar file.
      i. The makefile **shall** be able to build your source in a 32 bit Lubuntu VM.
      ii. The built program **shall** be able to run in a 32 bit x86 Lubuntu VM.