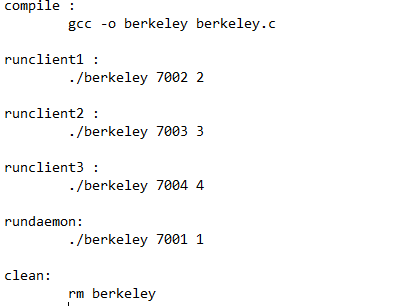
**Assignment 1:**

How to Compile: The following makefile is used to compile and run the code



**Type the following in the src directory:**

* make compile

This will generate one binary file named: berkeley

**Type the following command to delete the object files:**

* make clean

**To test the simulation, open 4 command line windows/prompts.**

For the first 3 command line prompts, run the clients by typing the following commands in each prompt respectively:

* make runclient1
* make runclient2
* make runclient3

This will activate the processes on ports 6666, 7777, and 8888 respectively.

For the fourth command line window/prompt, run the time synchronization process by typing the following:

* make rundaemon

Please note the following:

1. The port numbers for each of the servers are fixed, in other words, they must be 6666, 7777, and 8888.

2. For the synchronizing/client process, the process ID must have a value of 1: this is what determines that it is the synchronizing process. A process ID other than 1 is a server process.

3. A synchronizing process is the process that synchronizes the logical clock of all the other processes; the other processes are simply servers listening for a connection.

4. Once the synchronization/client process is done synchronizing the logical clocks of the listening servers, it closes on its own, but the other servers remain running and can accept new connection requests.

5. After running the synchronization/client process the first time, you can run it again to check the returned logical clock values from all the servers, they will be the SAME.

6. Each process starts with a pseudo-randomly generated logical clock value; in other words, the logical clocks of the processes/servers are not hard-coded, they change it every time they are run.

--X--