Report

Q3) Multithreaded client and server

Server Program:

- The server program spawns n (passed as a command line argument)threads and prints their thread ids. Each of the threads is launched into `worker_func` where it waits on the semaphore.
- The main program listens for client requests. Whenever a client request arrives, it increments a semaphore on which the server threads are waiting. This wakes up any of the threads currently waiting and assigns it the request.
- The main program maintains a queue of client_socket_fd through which the worker threads can receive and send information to the client.
- Whenever a worker thread is assigned a client, it pops the client_socket_fd from the queue and makes the necessary changes in the dictionary. It then sends the appropriate response to the client.

Client Program:

- The client program takes input as specified in the assignment pdf.
- It spawns m threads where m is the total number of client requests. Each thread is launched into `client_func` where it sleeps for the amount of time at which it is scheduled to arrive.
- After this, the client establishes a connection with the server and sends it the request.
- Then it waits for the response from server and prints the message received and then exits.

The `main` program of client side waits on all the client threads to return from the `client_func` and this finally ends the simulation.

The `main` program of server side is in an infinite loop and continues to wait for client requests. It can be exited by pressing ctrl + C.