Programming Assignment 1: Getting familiar with Sockets
CSE 310, Spring 2019
Instructor: Aruna Balasubramanian
Due date: 02/25/2019, 9.00pm

Sending information across the network requires creating a transport connection and then sending data over this connection. If you are using a TCP in the transport layer, then you need to create an explicit connection. If you are using UDP, then you do not need to. We will stick to TCP for now. Your goal is to write an application that creates a TCP connection and then write an application protocol on top of it.

Problem Description:

- The Addressbook service is a client-server application. A client sends a query message to the server containing an email address. The server responds to the client with a message that contains the full name which corresponds to the email address.

- The client submits query messages to the server. The query message contains an email address.

- The server runs on a known IP address (you can assume that the IP address is known) and listens on a well-known port number for client requests. When a request comes in, the server looks in a database to find the full name which corresponds to the email address and sends the full name to the requesting client.

- The following messages format is used for query and response:
  Message Type (1 byte)
  Length String (1 byte)
  Question/Response (<255 bytes)

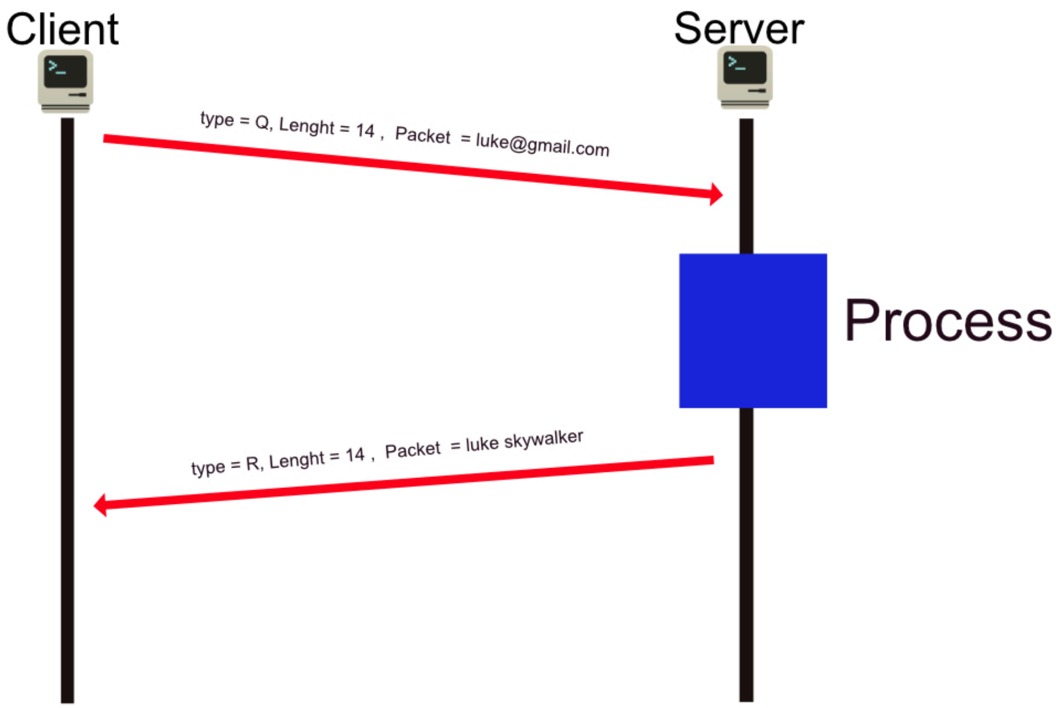
- Message Type is set to “Q” for queries, and “R” for responses.

- You can create the server and client on the same machine for ease of use, but make sure your code works when the server and client are on different machines.

- You get 10% extra credit if your server can handle multiple clients simultaneously.

A brief example in order to understand the assignment better:

Long time ago in a galaxy far far away there were two computers one of them as a client and the other one as a server. The client computer wanted to ask the server that who is the owner of the “luke@gmail.com" and the server answer according to this question was Luke Skywalker. The following image illustrates the packets:


Client
Server
type = Q, Lenght = 14 , Packet = luke@gmail.com
Process
type = R, Lenght = 14 , Packet = luke skywalker

Your Task:

1. Implement both the server and client of the Addressbook service. You can implement on C, C++, Java, or Python. Make sure that you handle error messages or other unexpected input correctly.

2. Write a summary describing your implementation, approach, and how to run your code.

3. Make sure to add comments to your code to get extra credit.