Networks Lab - Assignment 3

Group 25:
Tejas Khairnar: 180101081
Parth Bakare: 180101056

Compiling

```
gcc server.c -o server
gcc client.c -o client
```

Running

Starting the server:

Execute the following command taking the server port number as argument

```
./server <port number>
For example:  ./server 4200
```

Starting the client:

Execute the following command taking the server ip address and server port number as arguments.The server is bind to all available local interface ips.

```
./client <server ip address> <server port number>
```

NOTE: While running both server and client on a local machine use the IP of a loopback interface (127.0.0.1) found by using ifconfig command in the terminal and run the server and client in two different terminals

Properties

1. Server should run before client as client will connect to the server as soon as it runs.
2. Both client and server work on TCP sockets.
3. Server waits for TCP connection from client.

Concurency Check

Server starts a new child thread (using fork()) for every client so that multiple clients can be handled **concurrently**.

Working of the client-server

1. Once the client-server connection is setup, the client asks user to give input data which is then encoded using Base64 encoding where the maximum length os the message allowed is 1000 characters.
2. The client sends the encoded message to the server i.e the type 1 message.
3. After receiving the msg, server decodes the message and prints both encoded and decoded message and sends an "ACK" to the client i.e the type 2 message.
4. **Note:** The "ACK" sent by the server is also encoded, and is decoded by the client.
5. Server and client remain in the loop to send any number of messages which the client wishes.
6. When user doesn't wish to send more messages he can press any key other than 'y' key on his keyboard, client sends a type 3 message to server and connections is closed.

Screenshots

```
tejas@tejas-XPS-13-9380:~/Networks_LAB/Assignment-3$ gcc server.c -o server
tejas@tejas-XPS-13-9380:~/Networks_LAB/Assignment-3$ ./server 4200
SERVER WORKING

NEW CLIENT CONNECTION [127.0.0.1 : 7369] ESTABLISHED

Message received from client 127.0.0.1 : 7369
        Encoded Message: VGVqYXMgS2hhaXJuYXI=
        Decoded Message: Tejas Khairnar

Message received from client 127.0.0.1 : 7369
        Encoded Message: UGFydGggQmFrYXJl
        Decoded Message: Parth Bakare

CLIENT CONNECTION [127.0.0.1 : 7369] CLOSED
tejas@tejas-XPS-13-9380:~/Networks_LAB/Assignment-3$ gcc client.c -o client
tejas@tejas-XPS-13-9380:~/Networks_LAB/Assignment-3$ ./client 127.0.0.1 4200
Send message?
Press 'y' for YES or any other key for NO and press enter:
y
Enter the message to be sent
Tejas Khairnar
Recieved a message from server having IP 127.0.0.1 and port 4200
ACK
Send message?
Press 'y' for YES or any other key for NO and press enter:
y
Enter the message to be sent
Parth Bakare
Recieved a message from server having IP 127.0.0.1 and port 4200
ACK
Send message?
Press 'y' for YES or any other key for NO and press enter:
n
CONNCECTION CLOSED
```