**NA-1 Fall 2015 Project**

**Part 1. GENI / Socket programming Warm-up (50%)**

In this case we have used Python programming language.

We have developed and deployed s simple client and server programs on GENI. After the connection setup between the client and server, when we (Client side) type “Hello from Client-Mamun” (Shown in Fig.1), we can see the message on Server side (shown in Fig.2) and the server responses by the message “Hello from Server-Mamun” which is displayed on Client side (Fig.1). Then we typed few messages on Client side which all are displayed on Server (Fig.1 and Fig.2).

Finally, when we type “Bye from Client-Mamun” on Client side then the server replies “Bye from Server-Mamun” (displayed in Fig.1 and Fig.2).

***Client-Side Program:***

import socket

import socket

host='192.168.5.3'

port=4500

s=socket.socket()

s.connect((host,port))

message=raw\_input("enter -->")

while True:

if message == 'Hello from Client-Mamun':

s.send(message)

data=s.recv(1024)

print str(data)

message=raw\_input()

else:

s.send(message)

data=s.recv(1024)

print str(data)

message=raw\_input()

if message == 'Bye from Client-Mamun':

s.send(message)

data=s.recv(1024)

print str(data)

break

s.close()

***The Server side Program:***

import socket

import socket

host='192.168.5.3'

port=4500

s=socket.socket()

s.bind((host,port))

s.listen(1)

c,addr=s.accept()

while True:

data=c.recv(1024)

if data=='Bye from Client-Mamun':

c.send('Bye from Server-Mamun')

break

elif data=='Hello from Client-Mamun':

print str(data)

c.send('Hello from Server-Mamun')

else:

print str(data)

c.send(data)

c.close()

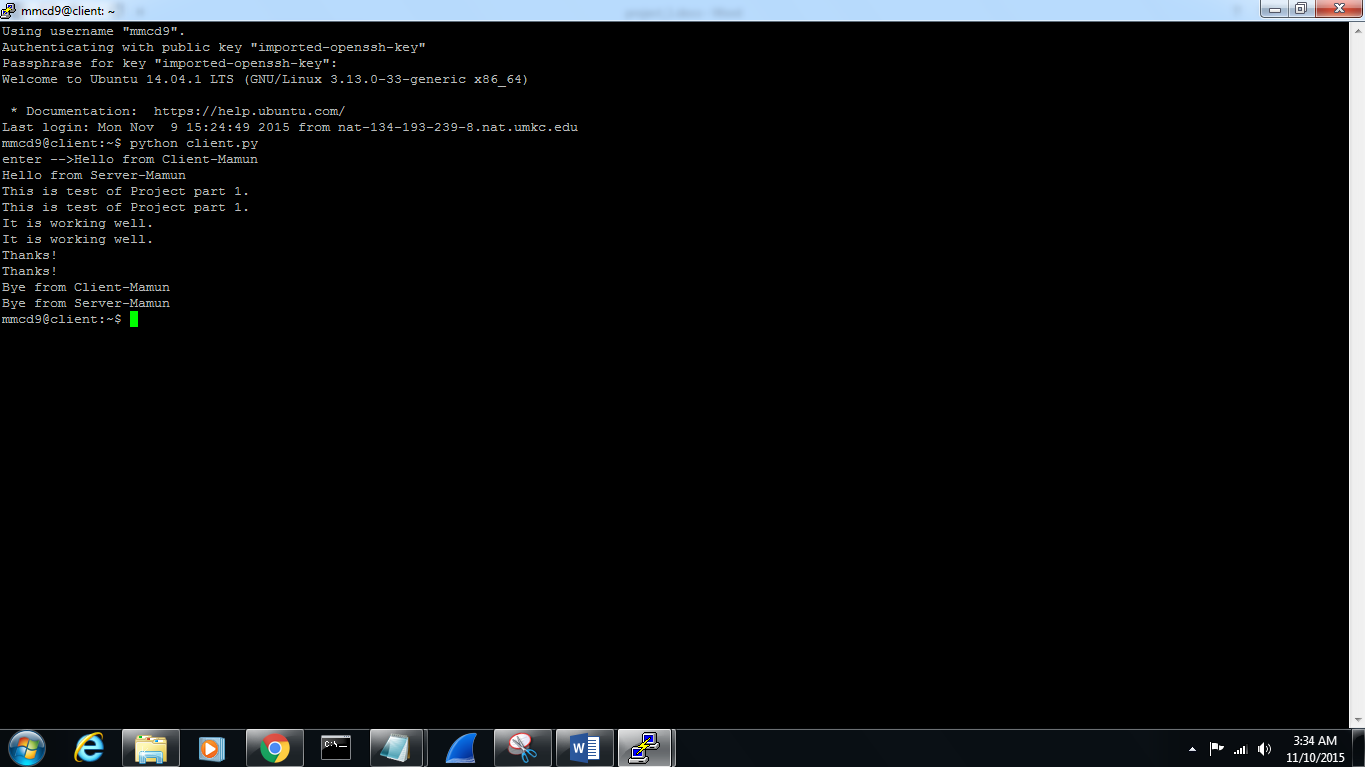


Fig. 1 Client

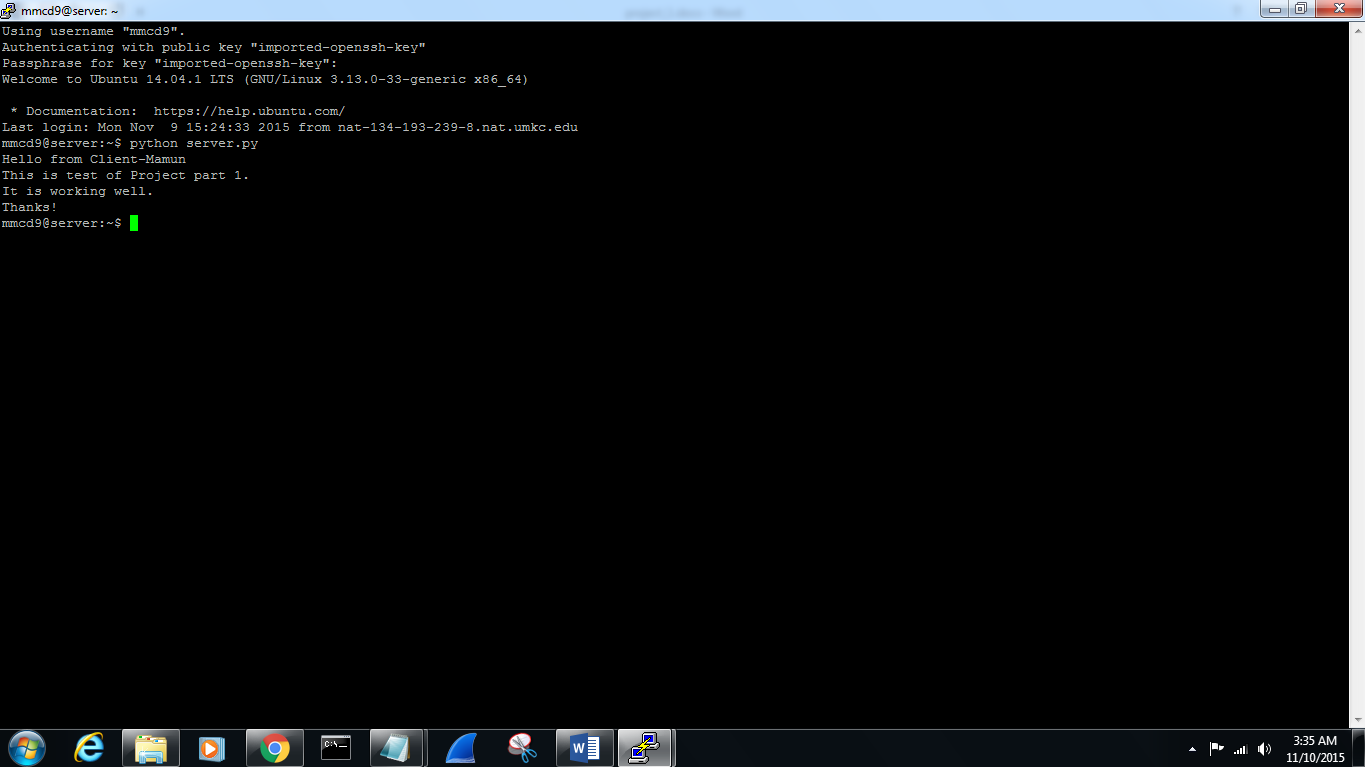


Fig. 2 Server

**Part II. Simple Chat Socket program (50%)**

1. We have developed a simple chat program using Python. The Client and Server are connected. The chat Server accepts a single client connection and displays everything the client types. The typed messages were shown in Fig.3 (Client Side) and these messages were displayed on Server side (Fig.4).

When the Client user types “quit”, both client and server end the program (shown in Fig.3 and Fig.4)

***The Client side program:***

import socket

import socket

host='192.168.5.3'

port=5000

s=socket.socket()

s.connect((host,port))

message=raw\_input("enter -->")

while True:

s.send(message)

if message=='quit':

break

data=s.recv(1024)

message=raw\_input()

s.close()

***The Server side program:***

import socket

import socket

host='192.168.5.3'

port=5000

s=socket.socket()

s.bind((host,port))

s.listen(1)

c,addr=s.accept()

print "Connection from:" + str(addr)

while True:

data=c.recv(1024)

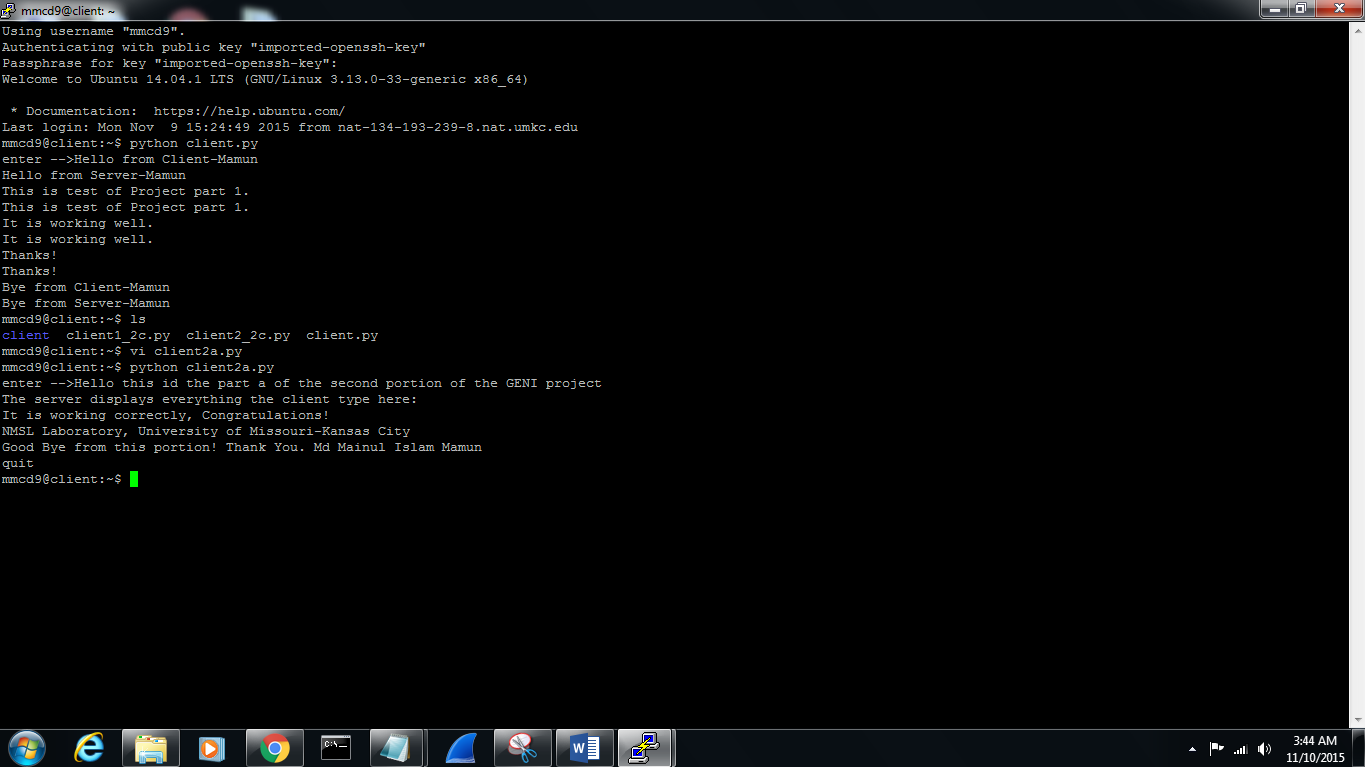
if data=='quit':

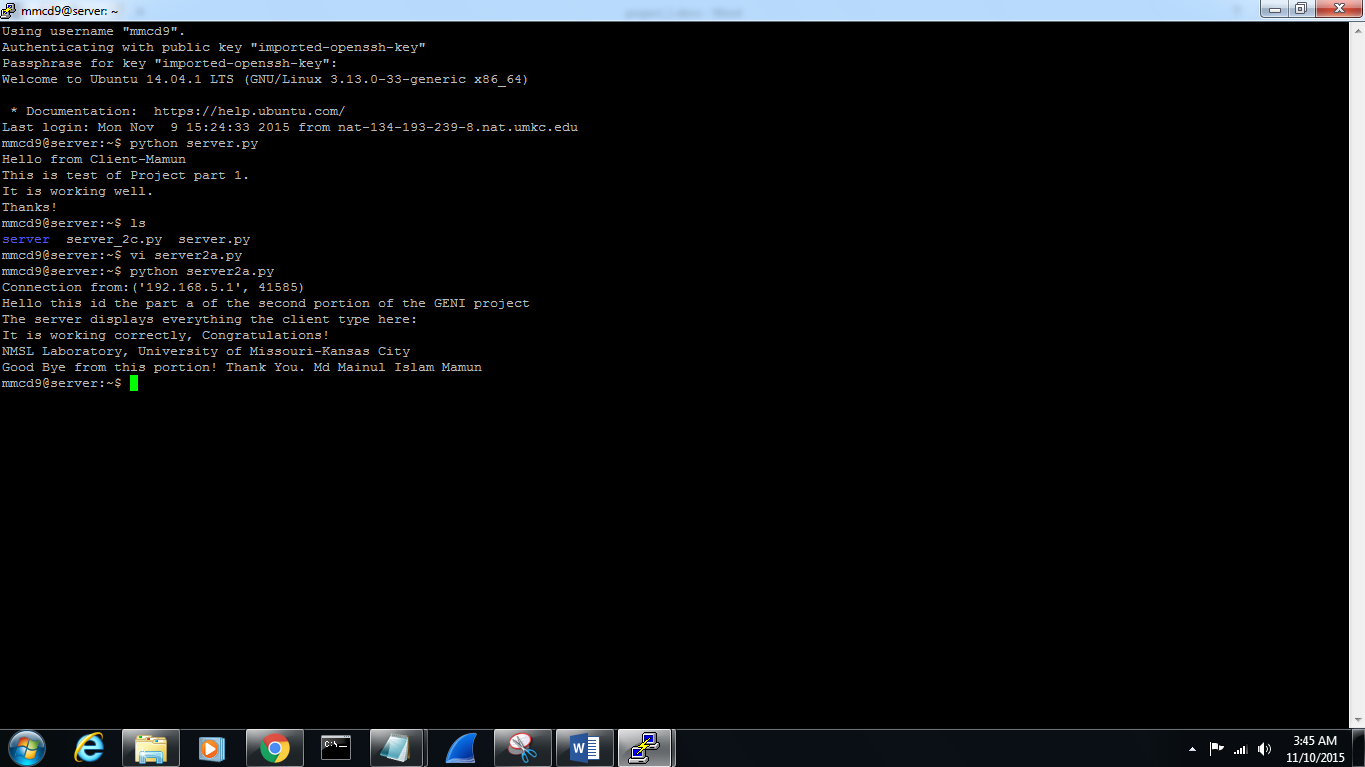
break

print data

c.send(data)

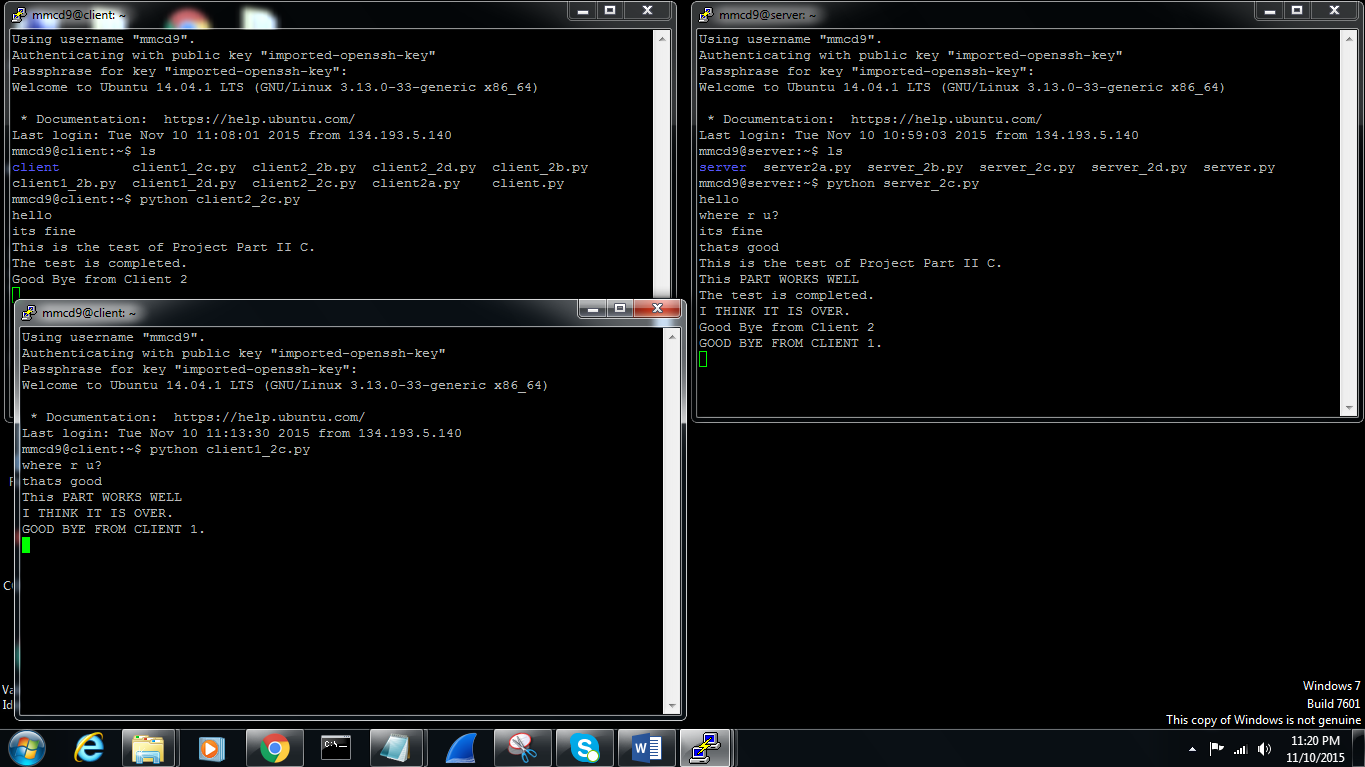
c.close()





b)

c)



d)

