Sam Lee

10/11/18

CPE 138

Professor Panzica

Socket Programming Lab #1

Part 1A:

Server Side Python File:

from socket import \*

serverPort = 12555

serverSocket = socket(AF\_INET, SOCK\_DGRAM)

serverSocket.bind(('', serverPort))

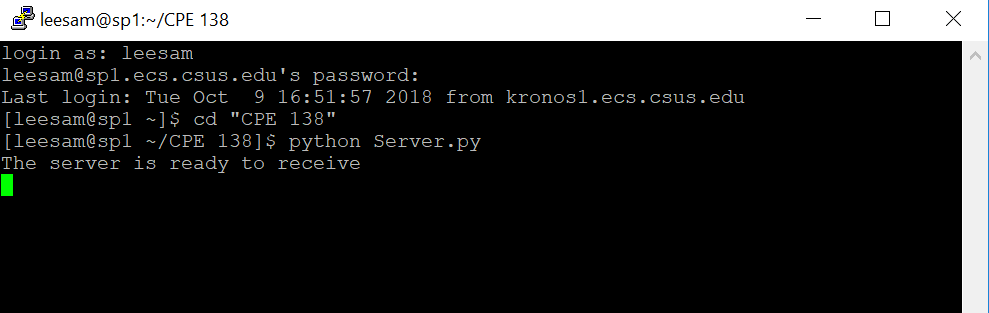
print("The server is ready to receive")

while True:

message, clientAddress = serverSocket.recvfrom(2048)

modifiedMessage = message.decode().upper()

serverSocket.sendto(modifiedMessage.encode(), clientAddress)



Client Side Python File:

from socket import \*

serverName = 'sp1.ecs.csus.edu'

serverPort = 12555

clientSocket = socket(AF\_INET, SOCK\_DGRAM)

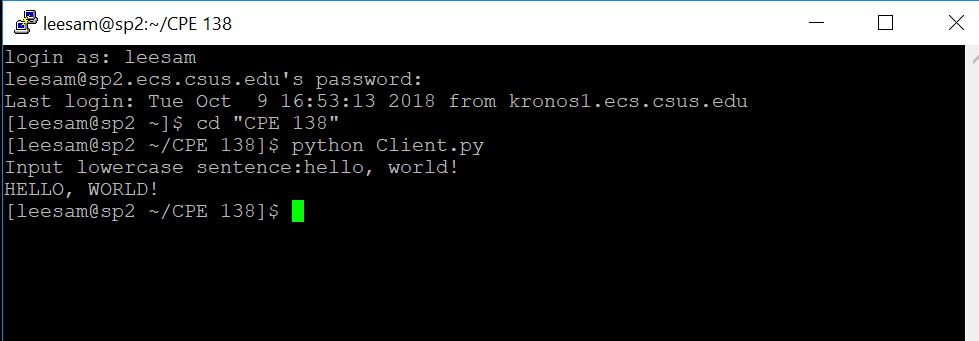
message = raw\_input('Input lowercase sentence:')

clientSocket.sendto(message.encode(), (serverName, serverPort))

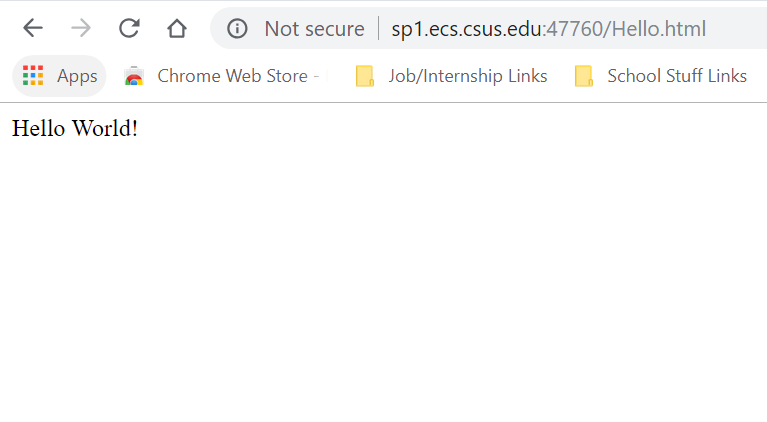
modifiedMessage, serverAddress = clientSocket.recvfrom(2048)

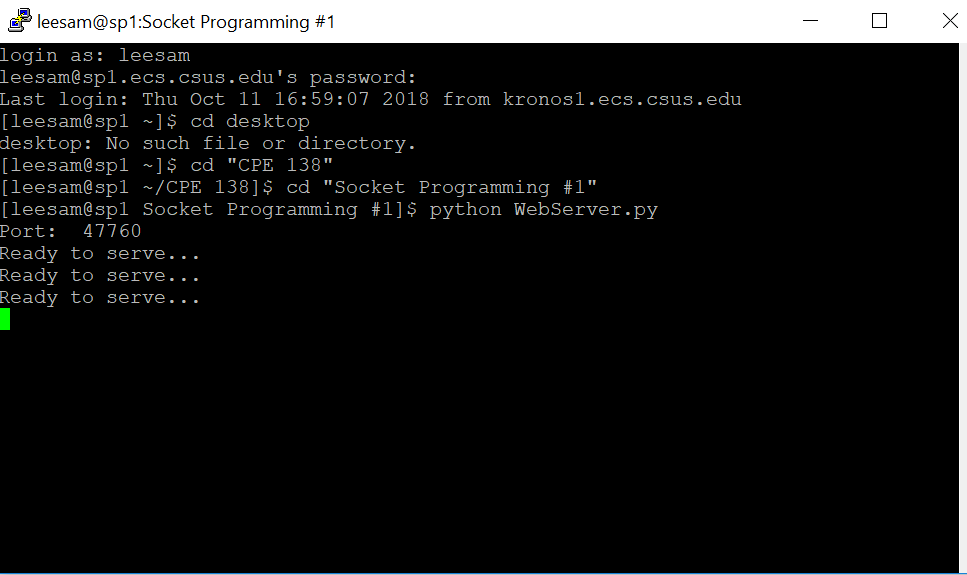
print modifiedMessage.decode()

clientSocket.close()



Part 1B:





WebServer Python File:

#import socket module

from socket import \*

import sys

serverSocket = socket(AF\_INET, SOCK\_STREAM)

#Prepare a server socket

port = 47760

serverSocket.bind(('', port))

serverSocket.listen(1)

print "Port: ", port

while True:

print 'Ready to serve...'

#Establish the connection

connectionSocket, addr = serverSocket.accept()

try:

message = connectionSocket.recv(1024)

filename = message.split()[1]

f = open(filename[1:])

outputdata = f.read()

f.close()

#Send one HTTP header line into socket

connectionSocket.send('HTTP/1.0 200 OK\r\n\r\n')

#Send the content of the requested file to the client

for i in range(0, len(outputdata)):

connectionSocket.send(outputdata[i].encode())

connectionSocket.send("\r\n".encode())

connectionSocket.close()

except IOError:

#Send response message for file not found

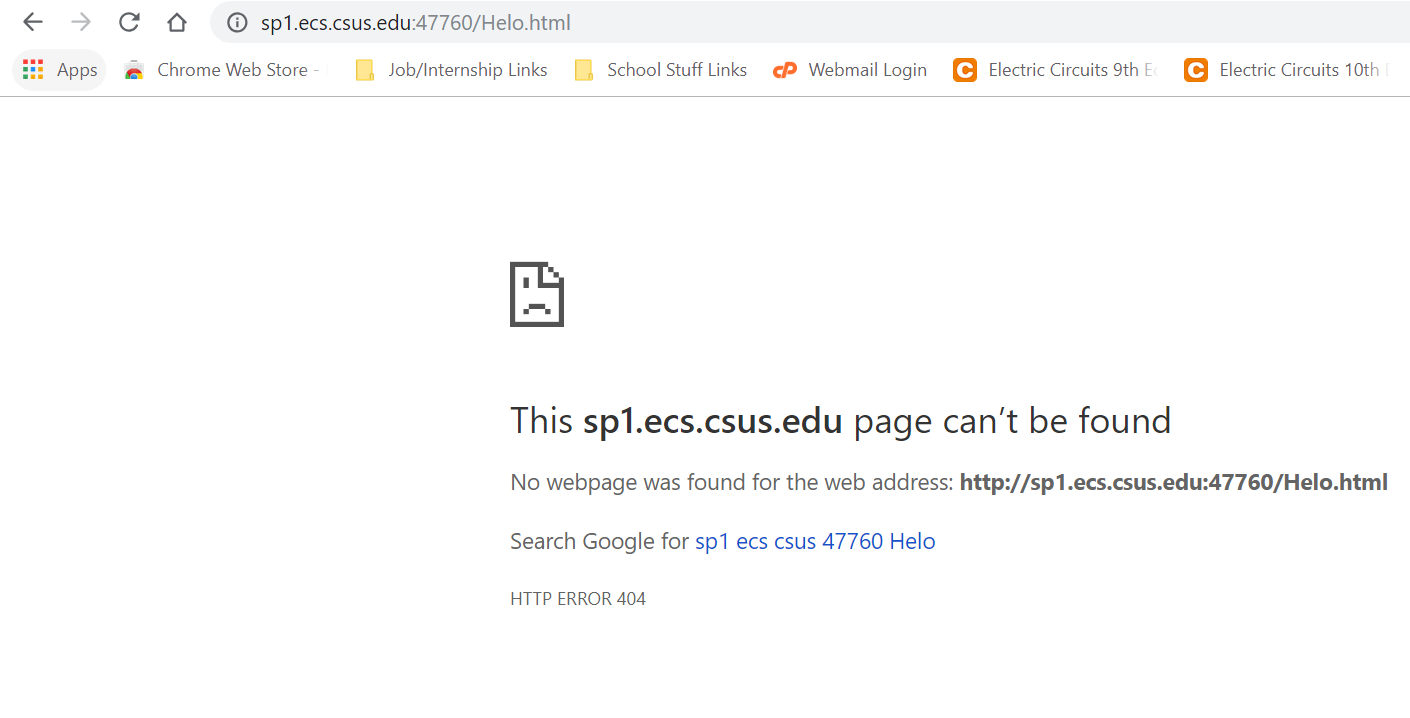
connectionSocket.send('HTTP/1.1 404 Not Found\r\n\r\n')

#Close client socket

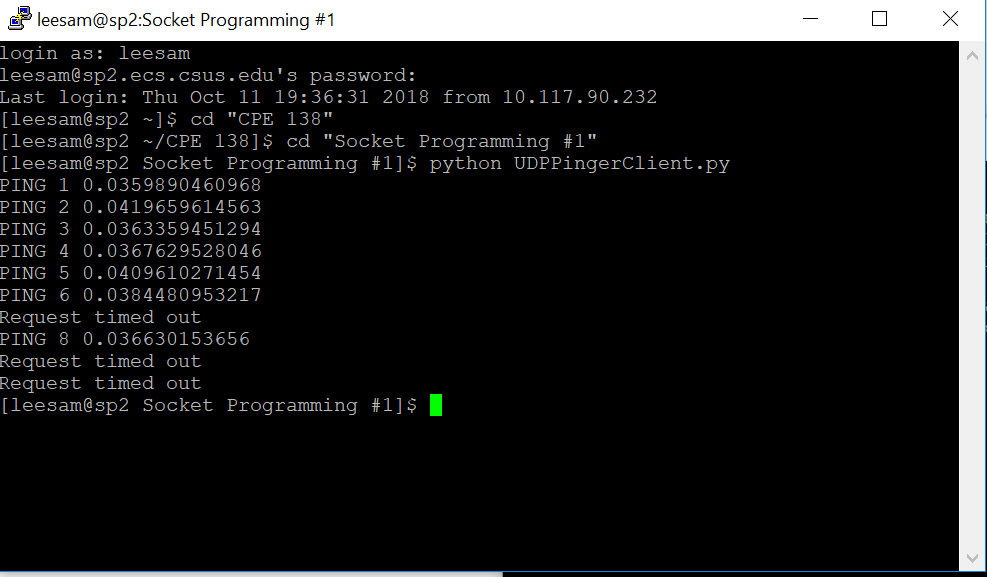
connectionSocket.close()

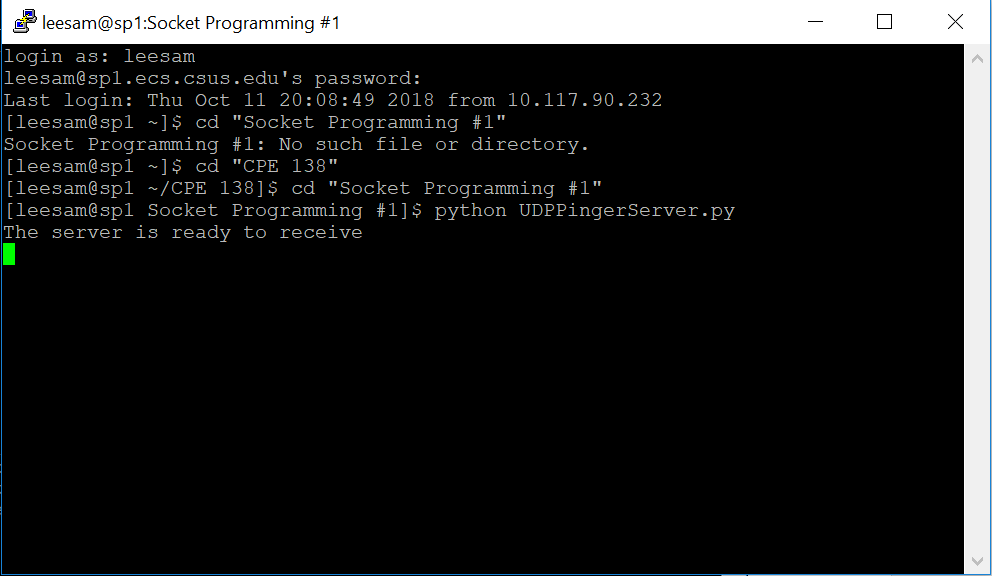
serverSocket.close()

sys.exit()#Terminate the program after sending the corresponding data



Part 1C:





UDPPingerClient Python File:

from socket import \*

import time

serverName = 'sp1.ecs.csus.edu'

serverPort = 12229

clientSocket = socket(AF\_INET, SOCK\_DGRAM)

clientSocket.settimeout(1)

message = 'ping'

counter = 1

while (counter <= 10):

startTime = time.time()

clientSocket.sendto(message.encode(), (serverName, serverPort))

try:

modifiedMessage, serverAddress = clientSocket.recvfrom(1024)

endTime = time.time()

print (modifiedMessage + ' ' + str(counter) + ' ' + str((endTime - startTime)))

except timeout:

print 'Request timed out'

counter = counter + 1

clientSocket.close()