Week 5. Socket Programming Project 1. Web Server Assignment - Due in Week5 Lab Class (5 points)

In this assignment, you will develop a simple **Web server** in Python that is capable of processing only **one request**. Specifically, your Web server will

(i) create a connection socket when contacted by a client (browser);

(ii) receive the HTTP request from this connection;

iii) parse the request to determine the specific file being requested;

(iv) get the requested file from the server's file system;

(v) create an HTTP response message consisting of the requested file preceded by header lines; and

(vi) send the response over the TCP connection to the requesting browser. If a browser requests a file that is not present in your server, your server should return a "404 Not Found" error message.

Code

Below you will find the skeleton code for the Web server. You are to complete the skeleton code. The places where you need to fill in code are marked with `#Fill in start` and `#Fill in end`. Each place may require one or more lines of code.

Running the Server

Put an HTML file (e.g., simpleWeb.html) in the same directory that the server is in. Run the server program. Determine the IP address of the host that is running the server (e.g., 128.238.251.26). From another host (PC, mobile phone, or iPad) in the same network, open a browser and provide the corresponding URL:

http://192.168.0.20:6789/simpleWeb.html

'HelloWorld.html' is the name of the file you placed in the server directory. Note also the use of the port number after the colon. You need to replace this port number with whatever port you have used in the server code. In the above example, we have used the port number 6789. The browser should then display the contents of HelloWorld.html. If you omit ":6789", the browser will assume port 80 and you will get the web page from the server only if your server is listening at port 80.

Then try to get a file that is not present at the server. You should get a "404 Not Found" message.

What to demonstrate to your tutor:

Group work, individually assessed.

You will demonstrate your server code to your tutor. Your tutor will verify that you actually receive the contents of the HTML file from the server. Note your tutor will ask specific questions to each of you to check your works and your understanding of the project.

Optional Exercises

In this project, you access your Web Server using a host within your own home /lab network. Are you able to access your web server from outside your home network? If not, can you find a solution so that any host in any network can access your web server? Explain your solution to your tutor.

Skeleton Python Code for the Web Server

```
#import socket module
from socket import *
serverSocket = socket(AF_INET, SOCK_STREAM)
#Prepare a sever socket
#Fill in start
#Fill in end
while True:
    #Establish the connection
    print('Ready to serve...')
    connectionSocket, addr =   #Fill in start              #Fill in end
    try:
        message =   #Fill in start          #Fill in end
        filename = message.split()[1]
        f = open(filename[1:])
        outputdata = #Fill in start       #Fill in end
        #Send one HTTP header line into socket
        #Fill in start
        #Fill in end
        #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
        #Fill in start
        #Fill in end
        #Close client socket
        #Fill in start
        #Fill in end
serverSocket.close()
sys.exit()#Terminate the program after sending the corresponding data
```