

CN Assignment 2 Report

Aditya Raj Jain - 2022037

Vedant Gupta - 2020261

Part A

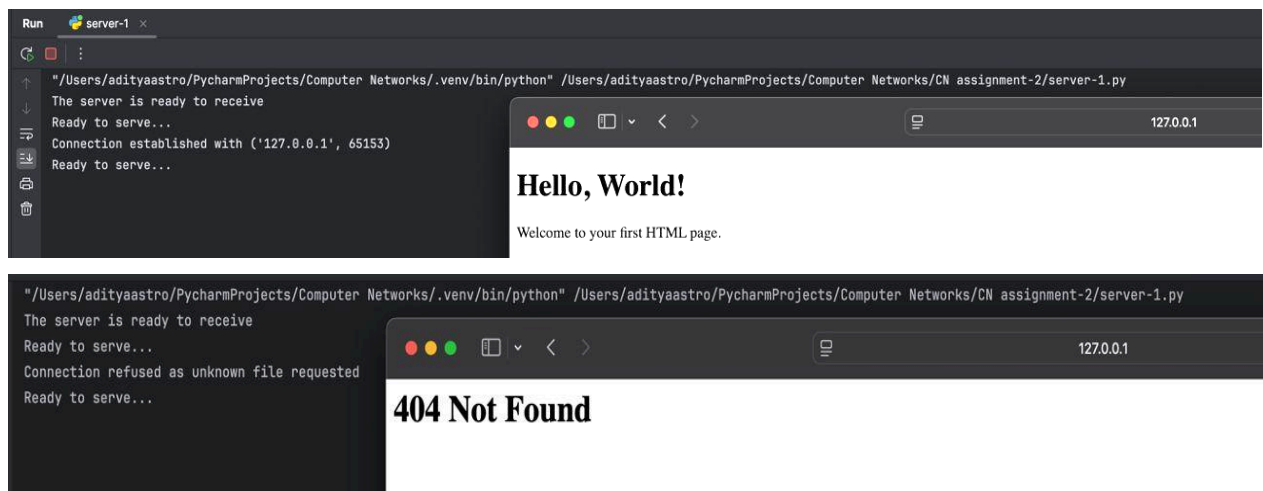
We designed a web server to handle one HTTP request at a time using TCP. It accepts a connection, parses the HTTP request, retrieves the requested file, and sends an HTTP response. The server responds with a 404 error message if the file is not found.

Assumptions:

1. The requested file is expected to exist in the server's directory.
2. The server returns a 404 Not Found status if the file does not exist.
3. The server listens on port 80, which is the default HTTP port.
4. A buffer size of 1024 bytes is used while decoding the incoming HTTP request.
5. The server is set to handle one connection at a time with a listen queue size of 1.

We utilized Python's `SOCK_STREAM` socket to establish a TCP connection, binding it to port 80 to listen for incoming requests. Upon a client's connection, the server accepts the connection and initiates the processing of the request.

The server receives the request message using the `recv` method and decodes it. It then extracts the filename from the HTTP request and attempts to open the corresponding file.



The client sends a request to the server for the file HelloWorld.html. The server parses this request, locates HelloWorld.html, and responds with the file along with a status code of 200 OK. The server returns a 404 Not Found error if the requested file is not found.

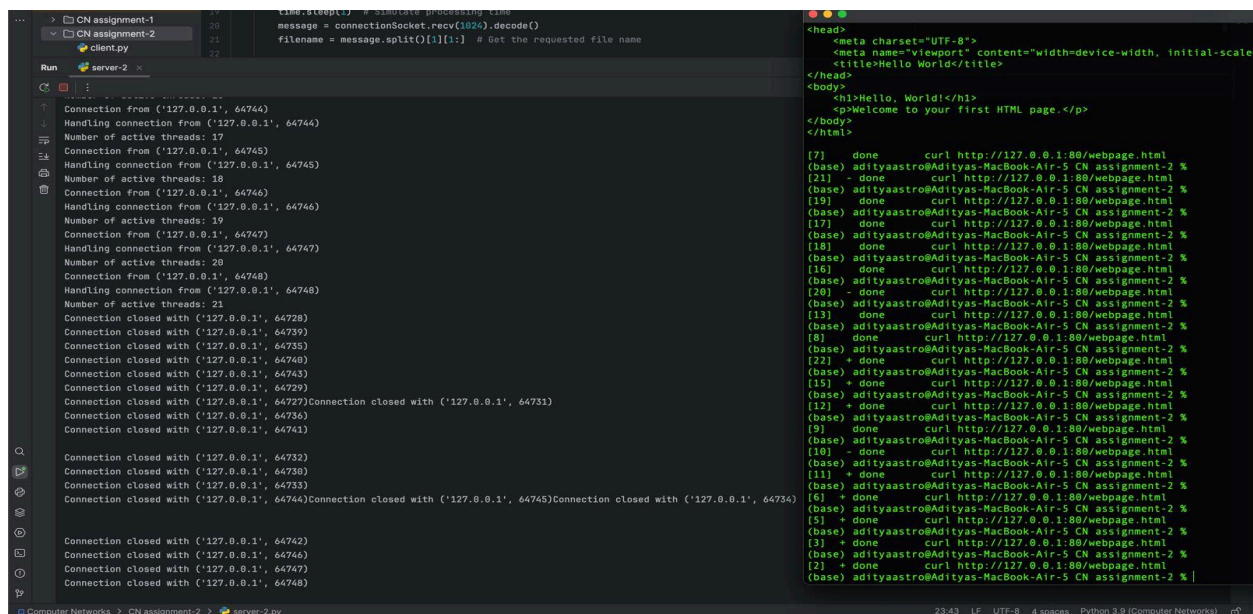
Part B

We have modified the initial code in part A to incorporate multithreading, allowing multiple users to retrieve the HTML file simultaneously. The server creates a new thread for each incoming request. To illustrate this functionality, we have set a timeout of 1 second.

When the server receives commands via “curl,” it processes all requests concurrently due to the timeout. We also display the number of active threads created to handle these simultaneous requests.

Assumptions:

1. A 1-second timeout has been set to ensure clear visibility of the outputs.
2. The curl command is used to test and send data to the server.
3. The listening queue has been increased to accommodate up to 5 connections in the queue.



```
...  > CN assignment-1  20  line.sleep(1) # simulate processing time
    > CN assignment-2  21  message = connectionSocket.recv(1024).decode()
    client.py         22  filename = message.split()[1][1:] # Get the requested file name
    server-2.py       23

Run  server-2.py
Connection from ('127.0.0.1', 64744)
Handling connection from ('127.0.0.1', 64744)
Number of active threads: 17
Connection from ('127.0.0.1', 64745)
Handling connection from ('127.0.0.1', 64745)
Number of active threads: 18
Connection from ('127.0.0.1', 64746)
Handling connection from ('127.0.0.1', 64746)
Number of active threads: 19
Connection from ('127.0.0.1', 64747)
Handling connection from ('127.0.0.1', 64747)
Number of active threads: 20
Connection from ('127.0.0.1', 64748)
Handling connection from ('127.0.0.1', 64748)
Number of active threads: 21
Connection closed with ('127.0.0.1', 64728)
Connection closed with ('127.0.0.1', 64739)
Connection closed with ('127.0.0.1', 64735)
Connection closed with ('127.0.0.1', 64740)
Connection closed with ('127.0.0.1', 64743)
Connection closed with ('127.0.0.1', 64729)
Connection closed with ('127.0.0.1', 64727)
Connection closed with ('127.0.0.1', 64731)
Connection closed with ('127.0.0.1', 64736)
Connection closed with ('127.0.0.1', 64741)

Connection closed with ('127.0.0.1', 64732)
Connection closed with ('127.0.0.1', 64730)
Connection closed with ('127.0.0.1', 64733)
Connection closed with ('127.0.0.1', 64744)
Connection closed with ('127.0.0.1', 64745)
Connection closed with ('127.0.0.1', 64746)
Connection closed with ('127.0.0.1', 64747)
Connection closed with ('127.0.0.1', 64748)

[7] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[21] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[19] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[17] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[18] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[16] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[20] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[13] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[8] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[22] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[15] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[9] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[12] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[6] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[5] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[3] + done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %
[2] done curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5: CN assignment-2 %

23:43 LF UTF-8 4 spaces Python 3.9.10 (Computer Networks)
```

In the above image, we issued 21 simultaneous requests, resulting in the creation of 21 active threads. The server then provides the HTML file to each thread before closing the corresponding connections.

```
server-2 x
Connection from ('127.0.0.1', 53115)
Number of active client threads: 6
Connection from ('127.0.0.1', 53114)
Number of active client threads: 7
Connection from ('127.0.0.1', 53116)
Number of active client threads: 8
File Not Found
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
File Not Found
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
Connection closed with ('127.0.0.1', 53116)
</head>
<body>
  <h1>Hello, World!</h1>
  <p>Welcome to your first HTML page.</p>
</body>
</html>
[8] + done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[5]  done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[4]  done      curl http://127.0.0.1:80/webpge.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[7] + done      curl http://127.0.0.1:80/webage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[3] - done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[2] - done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[1] - done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
[6] + done      curl http://127.0.0.1:80/webpage.html
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
```

In the above example, I issued 6 commands with the correct file name and 2 with incorrect file names. The server correctly responds with "File not found" for the 2 erroneous requests.

Part C

This part involves writing an HTTP client that interacts with the web server using a TCP connection to request and display files hosted by the server. The client sends an HTTP GET request to retrieve a file from the server and displays the server's response.

Assumptions:

1. The server is operational and accessible at the specified host and port, with the requested file in its directory.
2. The client uses the HTTP GET method to request files, and the server responds according to the HTTP protocol.
3. The server returns either the requested file or a 404 Not Found error message.
4. The client processes the response in 4096-byte chunks for large files.
5. As the output is displayed in the terminal, it appears in HTML rather than GUI format.

The client establishes a TCP connection to the server using the specified host and port with Python's `SOCK_STREAM` socket. After establishing the

connection, it sends an HTTP GET request formatted as per the HTTP/1.1 protocol, including the Host header.

It receives the server's response in 4096-byte chunks, printing the content until no more data is received, thereby effectively handling large files. Additionally, the client captures any exceptions that occur during the connection or transmission processes and displays relevant error messages to help identify and troubleshoot problems.

```
Run server-1
/Users/adityaastro/PycharmProjects/Computer Networks/.venv/bin/python /Users/adityaastro/PycharmProjects/Computer Networks/CN assignment-2/server-1.py
The server is ready to receive
Ready to serve...
Connection established with ('127.0.0.1', 50736)
Ready to serve...
Connection refused as unknown file requested
Ready to serve...

(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 % python client.py 127.0.0.1 80 webpage.html
HTTP/1.1 200 OK
Content-Type: text/html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Hello World</title>
</head>
<body>
  <h1>Hello, World!</h1>
  <p>Welcome to your first HTML page.</p>
</body>
</html>

(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 % python client.py 127.0.0.1 80 wrongfile.html
HTTP/1.1 404 Not Found

<html><head></head><body><h1>404 Not Found</h1></body></html>
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
```

The client was tested with both the single-threaded and multi-threaded server versions (Parts 1 and 2). It can handle multiple requests as long as the server supports concurrent connections.

```
Run server-2
/Users/adityaastro/PycharmProjects/Computer Networks/.venv/bin/python /Users/adityaastro/PycharmProjects/Computer Networks/CN assignment-2/server-2.py
The server is ready to receive
Connection from ('127.0.0.1', 52654)
Number of active client threads: 1
Connection closed with ('127.0.0.1', 52654)

(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 % python client.py 127.0.0.1 80 webpage.html
HTTP/1.1 200 OK
Content-Type: text/html

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Hello World</title>
</head>
<body>
  <h1>Hello, World!</h1>
  <p>Welcome to your first HTML page.</p>
</body>
</html>
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %

Connection from ('127.0.0.1', 53030)
Number of active client threads: 1
File Not Found
Connection closed with ('127.0.0.1', 53030)

(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 % python client.py 127.0.0.1 80 wrongfile.html
HTTP/1.1 404 Not Found

<html><head></head><body><h1>404 Not Found</h1></body></html>
(base) adityaastro@Adityas-MacBook-Air-5 CN assignment-2 %
```

In the screenshot above we can see it's working well for part 1 and 2.