Computer Networks Assignment 1
Part 1
19110207
Vishal Soni

Network Application using Socket Programming:

a) **Source Code**
[Link to github repo](#).

b) **Design document**

Please find detailed description of files as comments within code of the github files.

1) **Instantiating connection with the server.**

```
D:\Sem 6\Computer Networks\CN_Assignment1\server>python connect_server.py
socket created
socket binded to 4000
socket is listening
connection to: ('127.0.0.1', 51754)
```

2) **Running the client**

```
D:\Sem 6\Computer Networks\CN_Assignment1>cd client

D:\Sem 6\Computer Networks\CN_Assignment1\client>python connect_client.py
```

3) **CWD command:**
The cwd command gives us the current working directory.

```
D:\Sem 6\Computer Networks\CN_Assignment1\client>python connect_client.py
cwd
D:\Sem 6\Computer Networks\CN_Assignment1\server
```

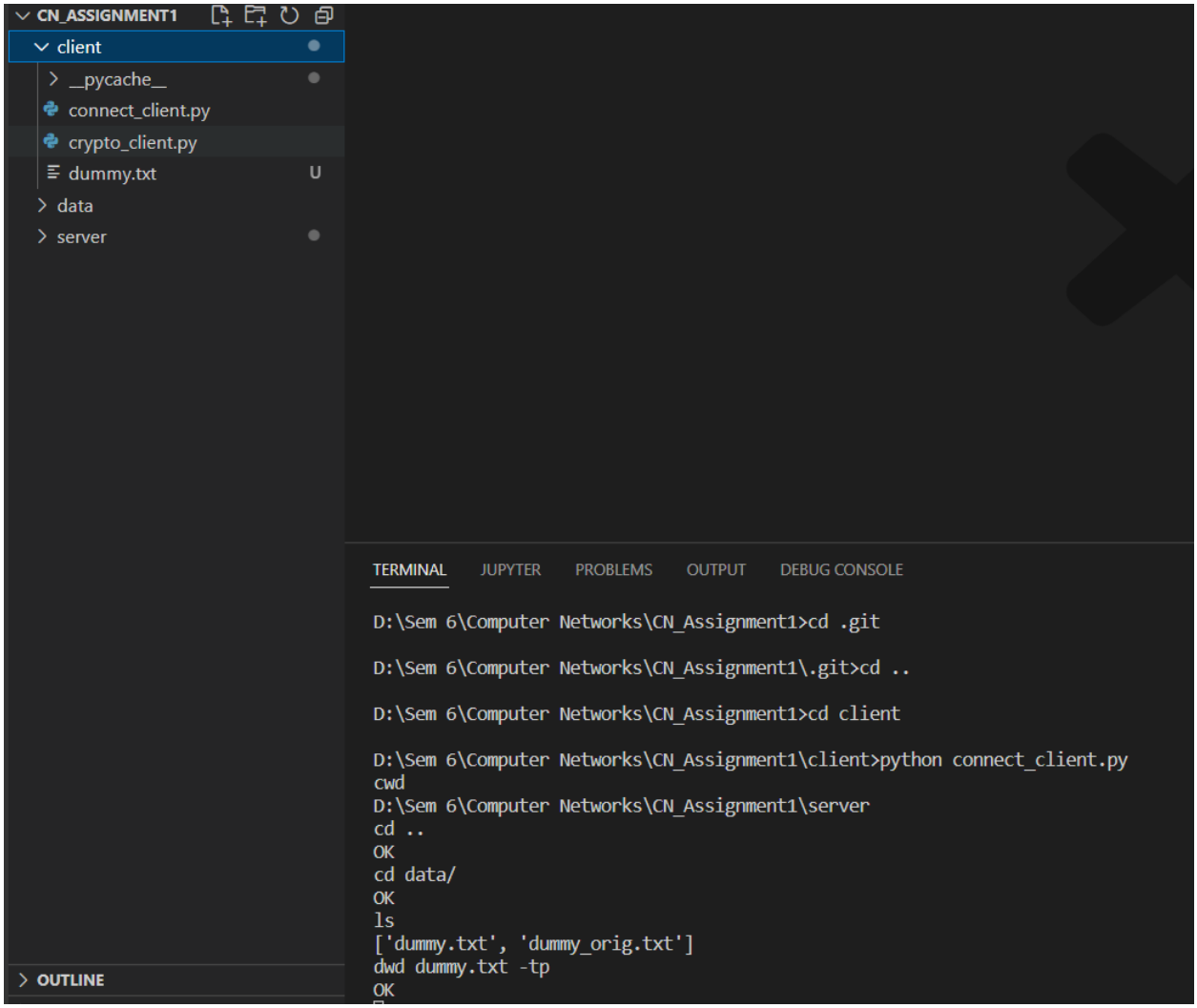
4) **The CD command takes us to the directory of choice:**

```
D:\Sem 6\Computer Networks\CN_Assignment1\client>python connect_client.py
cwd
D:\Sem 6\Computer Networks\CN_Assignment1\server
cd ..
OK
cd data/
OK
```

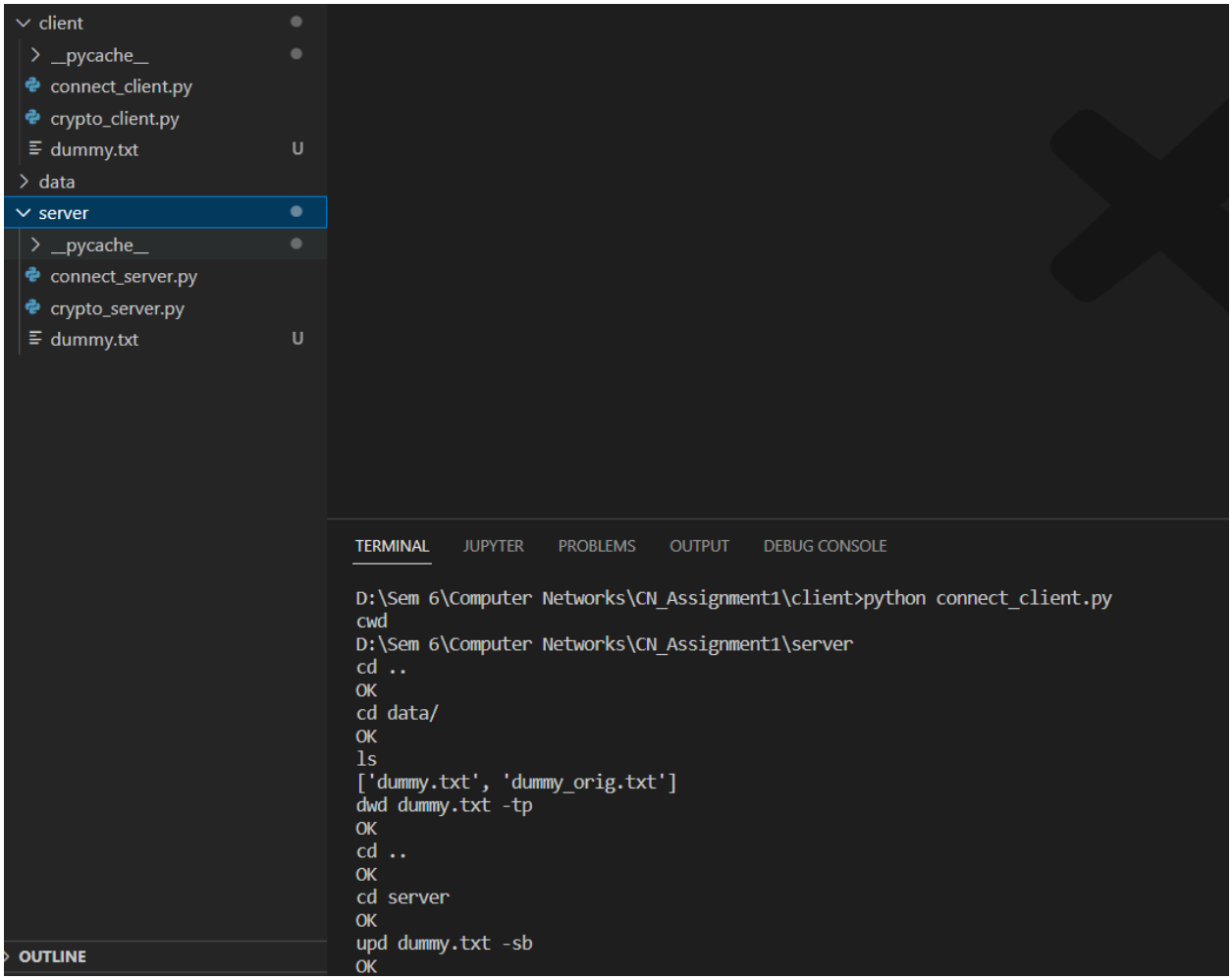
5) **The LS command lists the directories inside the current working directory**

```
D:\Sem 6\Computer Networks\CN_Assignment1\server
cd ..
OK
cd data/
OK
ls
['dummy.txt', 'dummy_orig.txt']
```

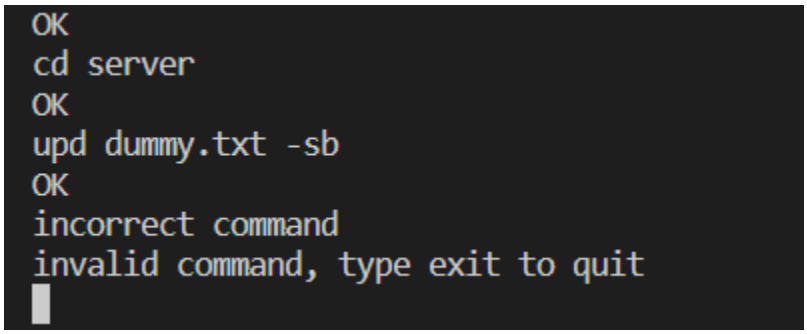
6) **The DWD command downloads dummy folder into the client directory**



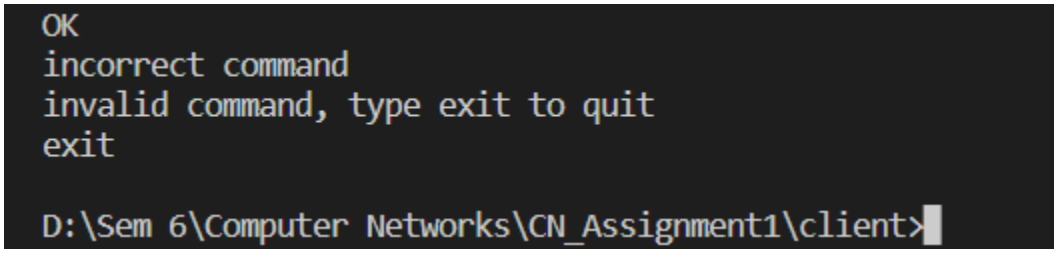
7) **The UPD command uploads the dummy.txt file from the client to the server:**



8) **Incorrect command shows error:**



9) **Exit terminates**



There will be client-side and server-side components. The code will support the five stated commands and look like a terminal. The client-side files cannot be controlled or accessed through the use of cwd, Is, or cd; only the server-side files may. We can select which files to download and where to upload them on the server by remotely going to various folders on the server. However, the client's system will only include one specific location for the downloaded files, and only those files can be uploaded to the server. The client has the option to encrypt files using plaintext, transpose or substitute encryption methods when downloading or uploading them.

We have three layers, application layer (terminal), encryption layer, and transport layer(networking). We have 4 files, **connect_server.py, connect_client.py, crypto_server.py and crypto_client.py**
The crypto files are responsible for encrypting and decrypting the files and the connect_server and connect_client, as the name suggests, are responsible for setting up socket connections of the server and the client.

c) **Wirshark dump and analysis**

Link to wireshark dump. For upload performed using transpose encoding from server to client.