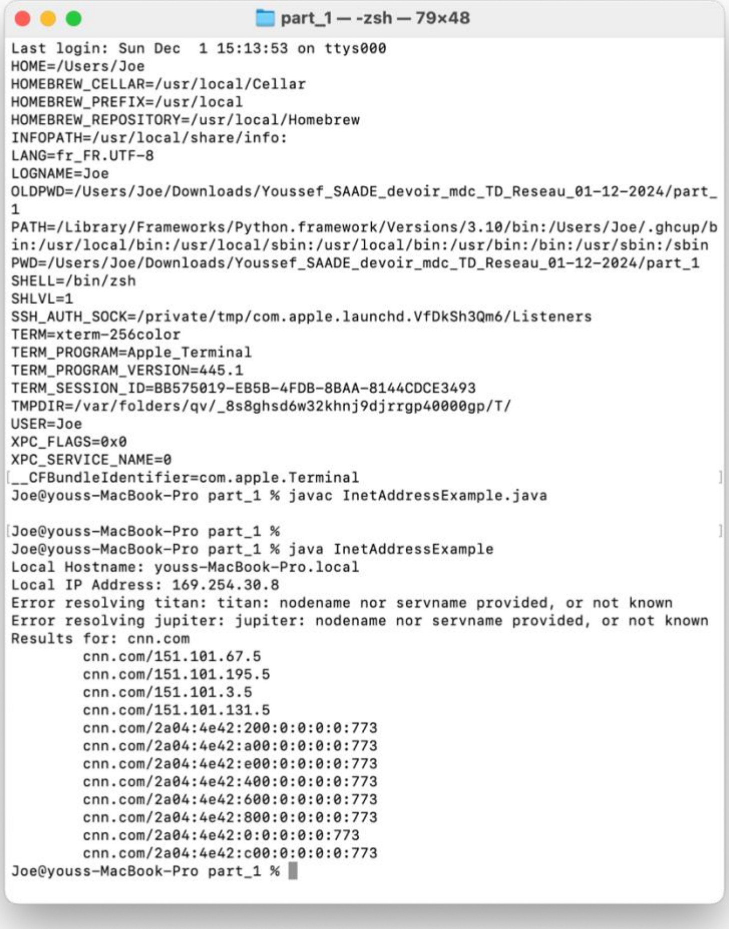
TD_NET

Youssef SAADE 01/12/2024

220958

Part 1 :



```
Last login: Sun Dec  1 15:13:53 on ttys000
HOME=/Users/Joe
HOMEBREW_CELLAR=/usr/local/Cellar
HOMEBREW_PREFIX=/usr/local
HOMEBREW_REPOSITORY=/usr/local/Homebrew
INFOPATH=/usr/local/share/info:
LANG=fr_FR.UTF-8
LOGNAME=Joe
OLDPWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_
1
PATH=/Library/Frameworks/Python.framework/Versions/3.10/bin:/Users/Joe/.ghcup/b
in:/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
PWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_1
SHELL=/bin/zsh
SHLVL=1
SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.VfDkSh3Qm6/Listeners
TERM=xterm-256color
TERM_PROGRAM=Apple_Terminal
TERM_PROGRAM_VERSION=445.1
TERM_SESSION_ID=BB575019-EB5B-4FDB-8BAA-8144CDCE3493
TMPDIR=/var/folders/qv/_8s8ghsd6w32khnj9djrrgp40000gp/T/
USER=Joe
XPC_FLAGS=0x0
XPC_SERVICE_NAME=0
__CFBundleIdentifier=com.apple.Terminal
Joe@youss-MacBook-Pro part_1 % javac InetAddressExample.java

Joe@youss-MacBook-Pro part_1 %
Joe@youss-MacBook-Pro part_1 % java InetAddressExample
Local Hostname: youss-MacBook-Pro.local
Local IP Address: 169.254.30.8
Error resolving titan: titan: nodename nor servname provided, or not known
Error resolving jupiter: jupiter: nodename nor servname provided, or not known
Results for: cnn.com
        cnn.com/151.101.67.5
        cnn.com/151.101.195.5
        cnn.com/151.101.3.5
        cnn.com/151.101.131.5
        cnn.com/2a04:4e42:200:0:0:0:0:773
        cnn.com/2a04:4e42:a00:0:0:0:0:773
        cnn.com/2a04:4e42:e00:0:0:0:0:773
        cnn.com/2a04:4e42:400:0:0:0:0:773
        cnn.com/2a04:4e42:600:0:0:0:0:773
        cnn.com/2a04:4e42:800:0:0:0:0:773
        cnn.com/2a04:4e42:0:0:0:0:0:773
        cnn.com/2a04:4e42:c00:0:0:0:0:773
Joe@youss-MacBook-Pro part_1 % 
```

Results

When running the program, the following output was observed:

1. **Local Hostname and IP Address:**
   - **Hostname:** youss-MacBook-Pro.local
   - **IP Address:** 169.254.30.8
   - Explanation: The hostname reflects the name of the machine on the local network, while the IP address is a "link-local" address automatically assigned when no DHCP server is available.
2. **Hostnames (titan and jupiter):**
   - **Error:** titan: nodename nor servname provided, or not known
   - **Error:** jupiter: nodename nor servname provided, or not known
   - Explanation: These hostnames could not be resolved because they are not valid domain names or mapped in the system's DNS or /etc/hosts file.
3. **Hostname (cnn.com):**
   - Resolved successfully to multiple IP addresses:
     - **IPv4 Addresses:** 151.101.67.5, 151.101.195.5, etc.
     - **IPv6 Addresses:** 2a04:4e42:200:0:0:0:0:773, etc.
   - Explanation: This demonstrates the proper resolution of a valid domain name using the DNS. The multiple IP addresses reflect load-balanced servers for cnn.com.

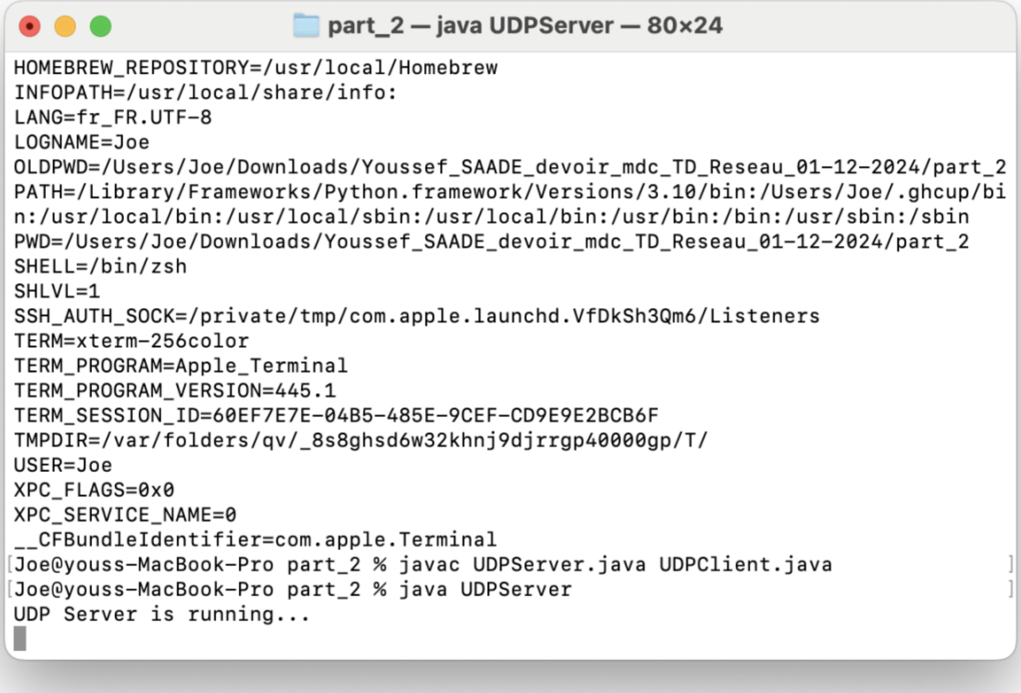
Conclusion

- **Local Machine:** The program successfully retrieved the local machine's hostname and IP address, showing how Java interacts with the network settings.
- **Unresolved Hostnames (titan and jupiter):** These failed because they are not configured in the DNS or the local system's /etc/hosts file.
- **Resolved Hostname (cnn.com):** The successful resolution of cnn.com shows Java's ability to query DNS servers and retrieve both IPv4 and IPv6 addresses for a valid domain name.

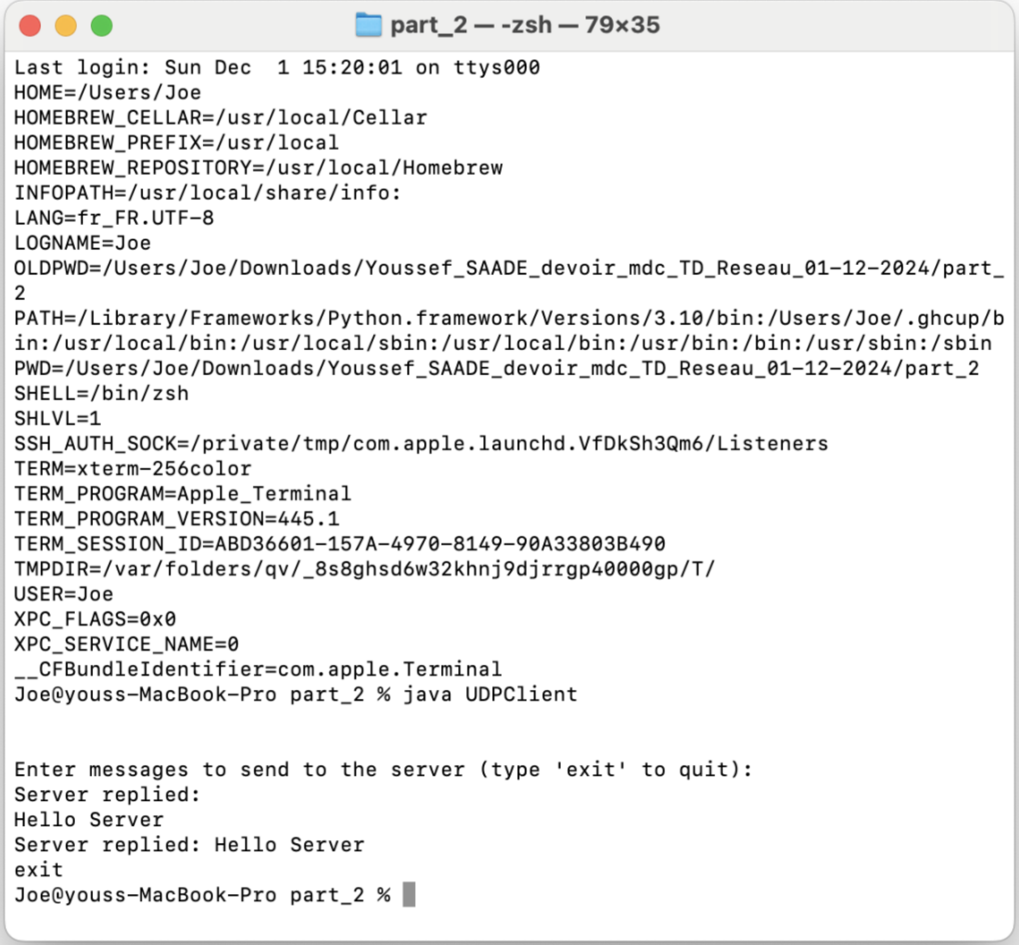
Potential Enhancements

- **Adding Hostnames:** To resolve titan and jupiter, they could be mapped to IP addresses in the system's /etc/hosts file.
- **Error Handling:** Additional error messages could be included in the program to suggest potential fixes for unresolved hostnames.

Part 2 :



```
HOMEBREW_REPOSITORY=/usr/local/Homebrew
INFOPATH=/usr/local/share/info:
LANG=fr_FR.UTF-8
LOGNAME=Joe
OLDPWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_2
PATH=/Library/Frameworks/Python.framework/Versions/3.10/bin:/Users/Joe/.ghcup/bi
n:/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
PWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_2
SHELL=/bin/zsh
SHLVL=1
SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.VfDkSh3Qm6/Listeners
TERM=xterm-256color
TERM_PROGRAM=Apple_Terminal
TERM_PROGRAM_VERSION=445.1
TERM_SESSION_ID=60EF7E7E-04B5-485E-9CEF-CD9E9E2BCB6F
TMPDIR=/var/folders/qv/_8s8ghsd6w32khnj9djrrgp40000gp/T/
USER=Joe
XPC_FLAGS=0x0
XPC_SERVICE_NAME=0
__CFBundleIdentifier=com.apple.Terminal
Joe@youss-MacBook-Pro part_2 % javac UDPServer.java UDPClient.java
Joe@youss-MacBook-Pro part_2 % java UDPServer
UDP Server is running...
█
```



```
Last login: Sun Dec  1 15:20:01 on ttys000
HOME=/Users/Joe
HOMEBREW_CELLAR=/usr/local/Cellar
HOMEBREW_PREFIX=/usr/local
HOMEBREW_REPOSITORY=/usr/local/Homebrew
INFOPATH=/usr/local/share/info:
LANG=fr_FR.UTF-8
LOGNAME=Joe
OLDPWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_
2
PATH=/Library/Frameworks/Python.framework/Versions/3.10/bin:/Users/Joe/.ghcup/b
in:/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
PWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_2
SHELL=/bin/zsh
SHLVL=1
SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.VfDkSh3Qm6/Listeners
TERM=xterm-256color
TERM_PROGRAM=Apple_Terminal
TERM_PROGRAM_VERSION=445.1
TERM_SESSION_ID=ABD36601-157A-4970-8149-90A33803B490
TMPDIR=/var/folders/qv/_8s8ghsd6w32khnj9djrrgp40000gp/T/
USER=Joe
XPC_FLAGS=0x0
XPC_SERVICE_NAME=0
__CFBundleIdentifier=com.apple.Terminal
Joe@youss-MacBook-Pro part_2 % java UDPClient


Enter messages to send to the server (type 'exit' to quit):
Server replied:
Hello Server
Server replied: Hello Server
exit
Joe@youss-MacBook-Pro part_2 % █
```

Code Implementation

1. **UDP Server (UDPServer.java):**
   - Listens on port 9876.
   - Receives messages from the client, processes them, and echoes them back.
   - If the message is "exit", the server shuts down.
2. **UDP Client (UDPClient.java):**
   - Sends messages entered by the user to the server.
   - Receives and displays the server's response.

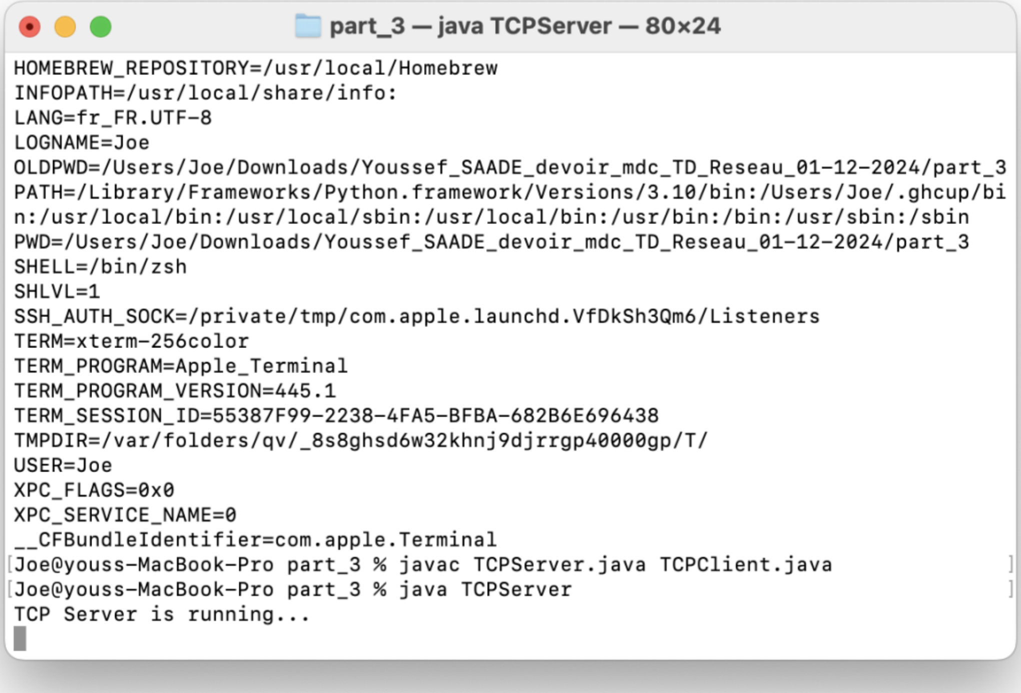
Explanation

1. **UDP Characteristics:**
   - **Connectionless Protocol:** Unlike TCP, there is no persistent connection between the client and server.
   - **Datagrams:** Messages are sent as independent packets (datagrams) with no guaranteed delivery or order.
2. **Message Flow:**
   - The client sends a message to the server.
   - The server receives the message, processes it, and sends the same message back.
   - The client displays the server's response.
3. **Key Observations:**
   - The server and client communicated successfully over UDP.
   - Multiple messages were handled without any connection setup.
   - The exit message allowed for clean termination of the server.

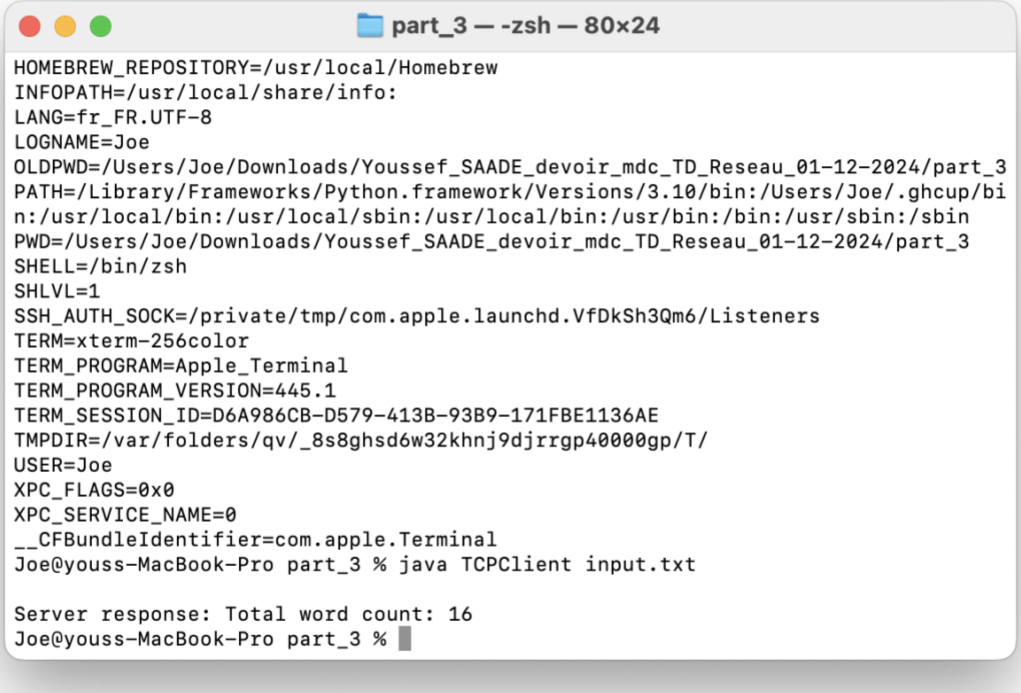
Conclusion

- The UDP Echo Server and Client were successfully implemented.
- The program demonstrated the characteristics of the UDP protocol:
  - **Low overhead:** Messages are sent without the need for a handshake.
  - **Stateless communication:** Each message is processed independently.
- The implementation showed how UDP is suitable for simple, fast, and connectionless communication.

part 3 :



```
HOMEBREW_REPOSITORY=/usr/local/Homebrew
INFOPATH=/usr/local/share/info:
LANG=fr_FR.UTF-8
LOGNAME=Joe
OLDPWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_3
PATH=/Library/Frameworks/Python.framework/Versions/3.10/bin:/Users/Joe/.ghcup/bi
n:/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
PWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_3
SHELL=/bin/zsh
SHLVL=1
SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.VfDkSh3Qm6/Listeners
TERM=xterm-256color
TERM_PROGRAM=Apple_Terminal
TERM_PROGRAM_VERSION=445.1
TERM_SESSION_ID=55387F99-2238-4FA5-BFBA-682B6E696438
TMPDIR=/var/folders/qv/_8s8ghsd6w32khnj9djrrgp40000gp/T/
USER=Joe
XPC_FLAGS=0x0
XPC_SERVICE_NAME=0
__CFBundleIdentifier=com.apple.Terminal
Joe@youss-MacBook-Pro part_3 % javac TCPServer.java TCPClient.java
Joe@youss-MacBook-Pro part_3 % java TCPServer
TCP Server is running...
```



```
HOMEBREW_REPOSITORY=/usr/local/Homebrew
INFOPATH=/usr/local/share/info:
LANG=fr_FR.UTF-8
LOGNAME=Joe
OLDPWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_3
PATH=/Library/Frameworks/Python.framework/Versions/3.10/bin:/Users/Joe/.ghcup/bi
n:/usr/local/bin:/usr/local/sbin:/usr/local/bin:/usr/bin:/bin:/usr/sbin:/sbin
PWD=/Users/Joe/Downloads/Youssef_SAADE_devoir_mdc_TD_Reseau_01-12-2024/part_3
SHELL=/bin/zsh
SHLVL=1
SSH_AUTH_SOCK=/private/tmp/com.apple.launchd.VfDkSh3Qm6/Listeners
TERM=xterm-256color
TERM_PROGRAM=Apple_Terminal
TERM_PROGRAM_VERSION=445.1
TERM_SESSION_ID=D6A986CB-D579-413B-93B9-171FBE1136AE
TMPDIR=/var/folders/qv/_8s8ghsd6w32khnj9djrrgp40000gp/T/
USER=Joe
XPC_FLAGS=0x0
XPC_SERVICE_NAME=0
__CFBundleIdentifier=com.apple.Terminal
Joe@youss-MacBook-Pro part_3 % java TCPClient input.txt

Server response: Total word count: 16
Joe@youss-MacBook-Pro part_3 %
```

Code Implementation

1. **TCP Server (TCPServer.java):**
    - Listens on port 6789.
    - Accepts client connections and reads text line by line.
    - Counts the total number of words received until the termination signal (xxxx) is sent by the client.
    - Sends the total word count back to the client.
    - Handles one client connection at a time.
2. **TCP Client (TCPClient.java):**
    - Reads a text file specified as a command-line argument.
    - Sends the file's contents line by line to the server.
    - Appends a termination signal (xxxx) after the file content.
    - Displays the server's response (total word count).
    -

Explanation

1. **TCP Characteristics:**
    - **Connection-Oriented Protocol:** A connection is established between the client and server before data transmission.
    - **Reliable Communication:** Data is sent and received in the correct order, ensuring accurate word counting.
2. **Message Flow:**
    - The client reads the file input.txt and sends its contents line by line.
    - The server receives the lines, counts the words, and sends the total count back to the client.
    - The termination signal xxxx ensures the server knows when the client has finished sending data.
3. **Key Observations:**
    - The server correctly handled client requests and returned the word count.
    - The client successfully transmitted file data and displayed the server's response.

Conclusion

- The TCP Server and Client were successfully implemented as per the requirements.
- The program demonstrated the characteristics of the TCP protocol:
    - **Reliable and ordered data transfer.**
    - **Persistent connection for the duration of communication.**
- The total word count of 16 matched the expected result based on the file content.