LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

Part-1 Socket Programming

Socket programming is a widely used concept in networking and provides communication mechanisms between two computers using HTTP/TCP. A socket is an endpoint of a two-way communication link between two programs running on the network. When the connection is made, the server creates a socket object on its end of the communication. The client and server can now communicate by writing to and reading from the socket.

HTTP GET and PUT Methods are two most commonly used methods for a request-response between a client and server are: GET and POST.

- **GET** - Requests data from a specified resource
- **POST** - Submits data to be processed to a specified resource

GET

In GET method, the clients connects to the server using a TCP connection. After this is done, a **filename** with **filepath** is sent as a request to the server, requesting for the contents of the file to be shown. On the server's side, the availability of the file is checked first by the server and then the content is sent if it is available along with a status message "**200 OK**". If the content is not available then a message "**404 Not Found**" is sent to the client.

PUT

In the PUT method, the clients gets connected to the server using TCP connection. The client then checks for the availability and if the file exists, the file is sent to the server. The server in turn saves the file and sends "**200 OK**" message to the client, if saved.

Execution Steps for GET:

Client:

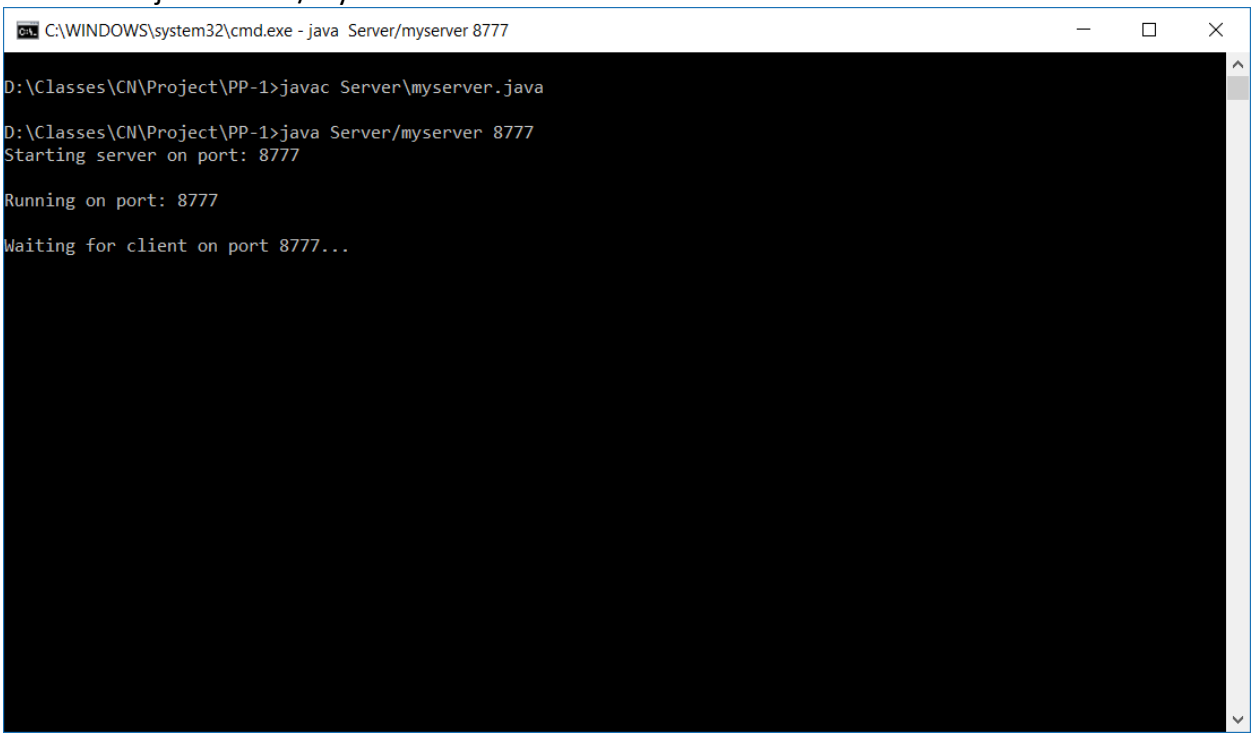
Compiling - javac Client\myclient.java

Execution - java Client/myclient localhost 8777 GET D:\Classes\CN\Project\PP-1\Server\getdata.txt

Server:

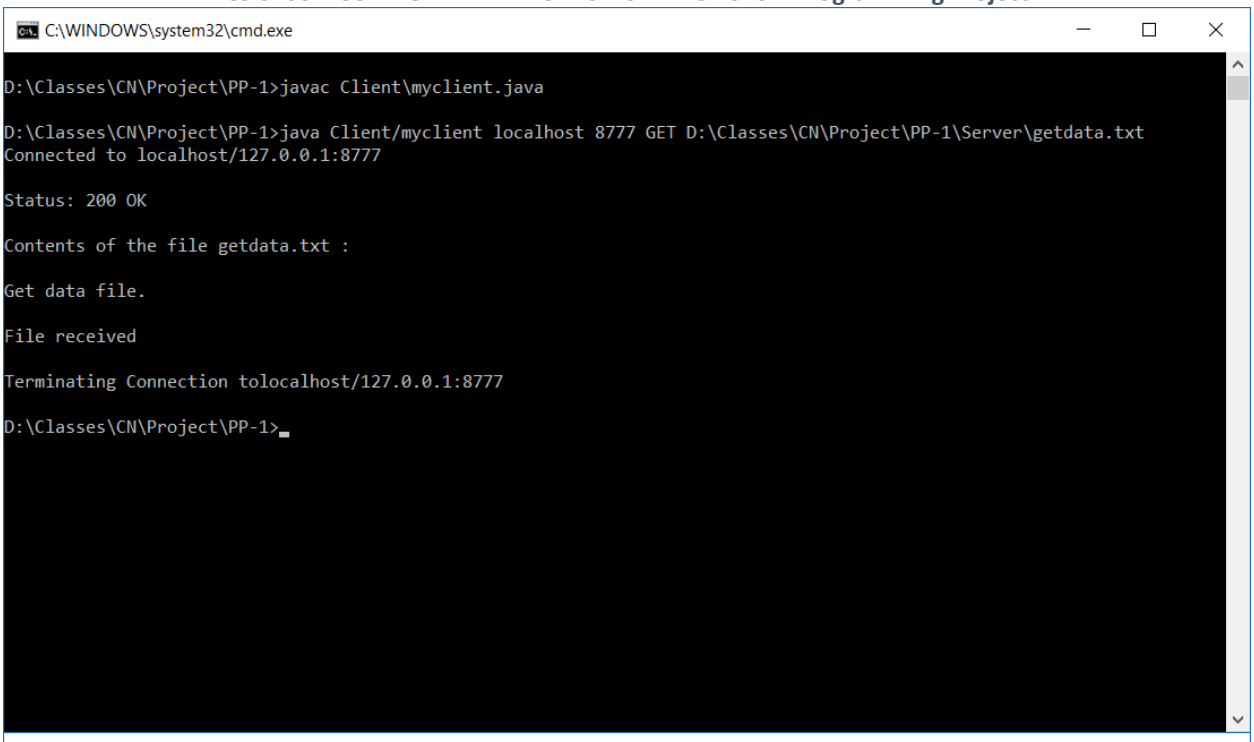
Compiling - javac \Server\myserver.java

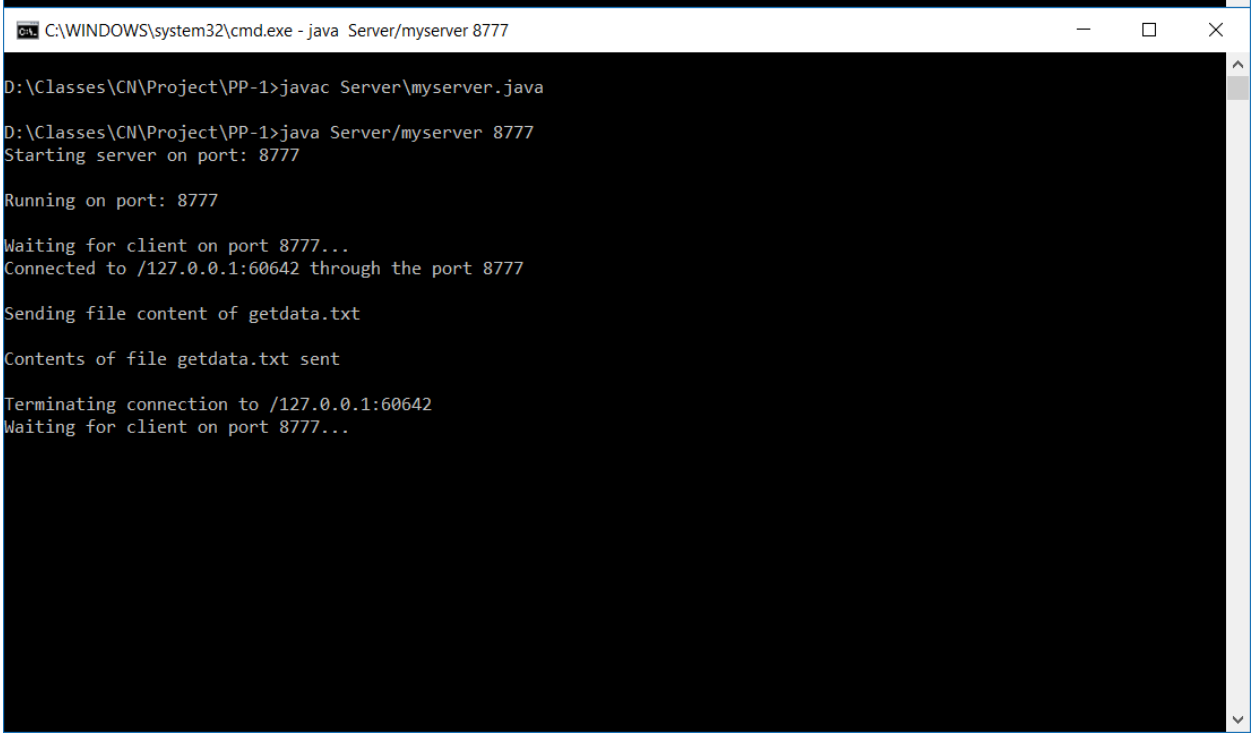
Execution - java Server/myserver 8777



LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

ITCS 6166 – COMPUTER NETWORKS – SPRING 2016 – Programming Project 1





Execution Steps for PUT:

Client:

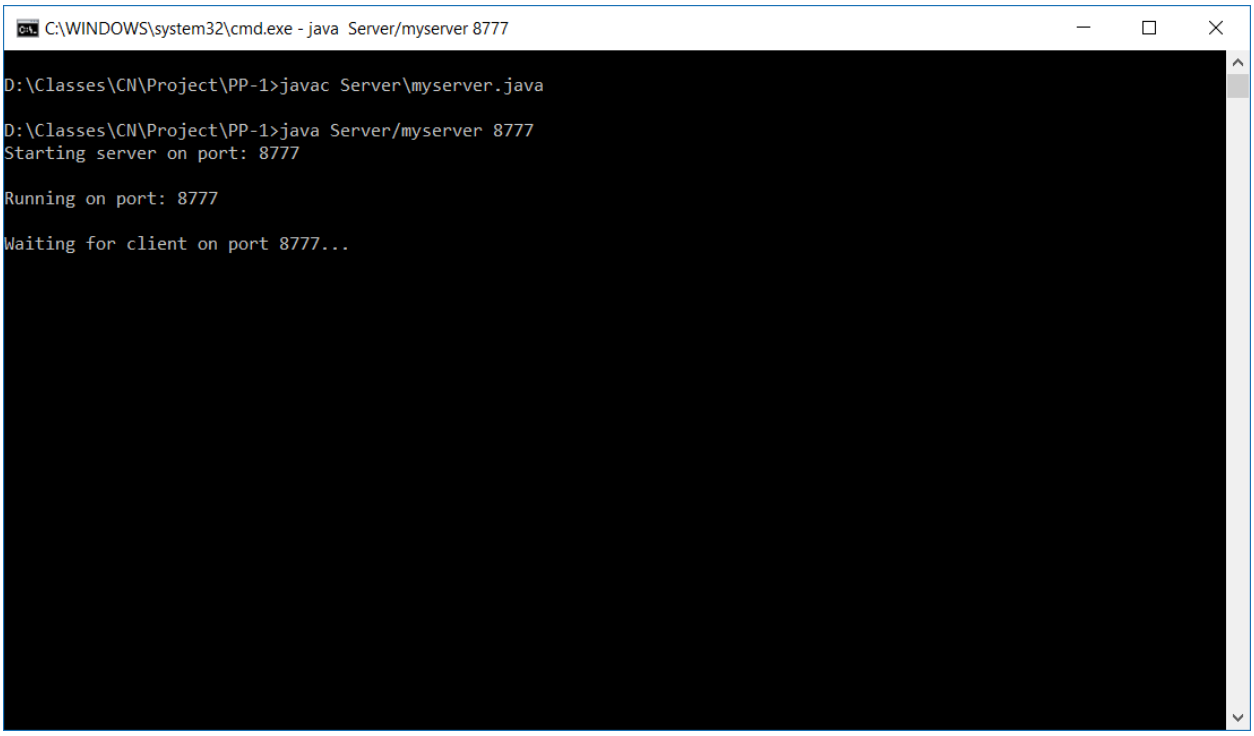
Compiling - javac Client\myclient.java

Execution - java Client/myclient localhost 8777 PUT
D:\Classes\CN\Project\Socket_Programming_1\Client\putdata.txt

Server:

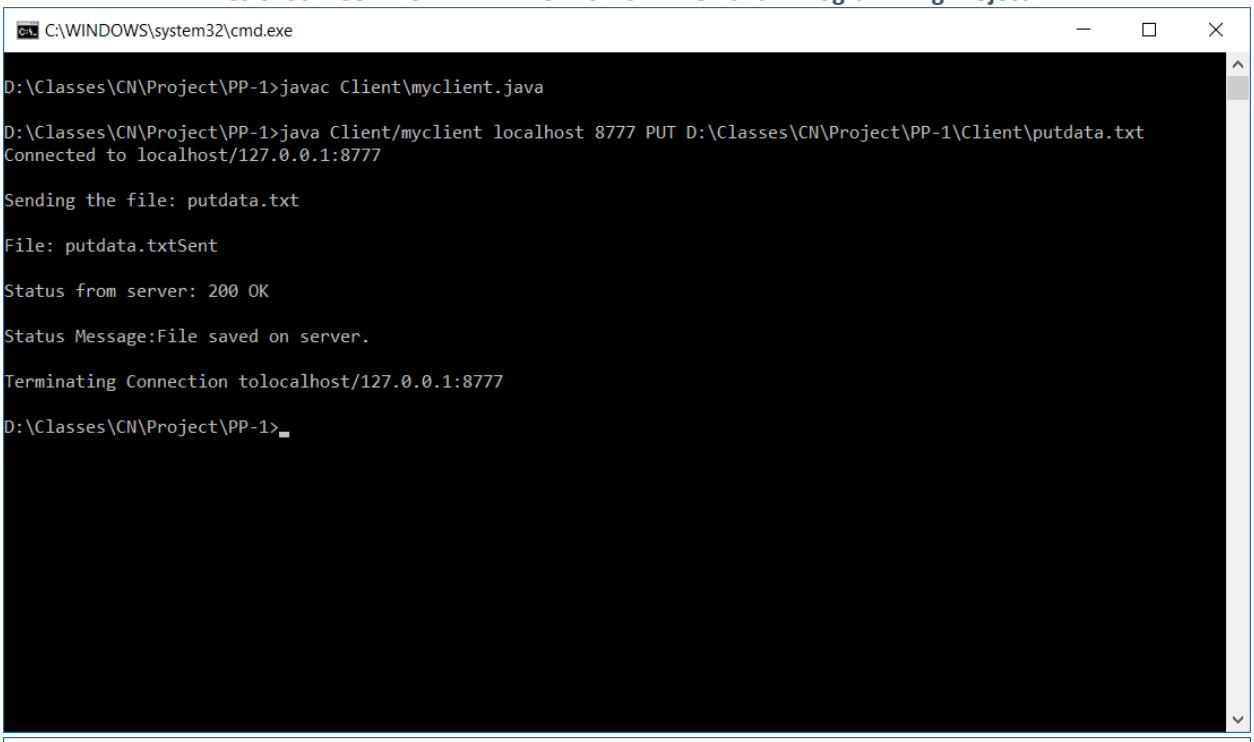
Compiling - javac \Server\myserver.java

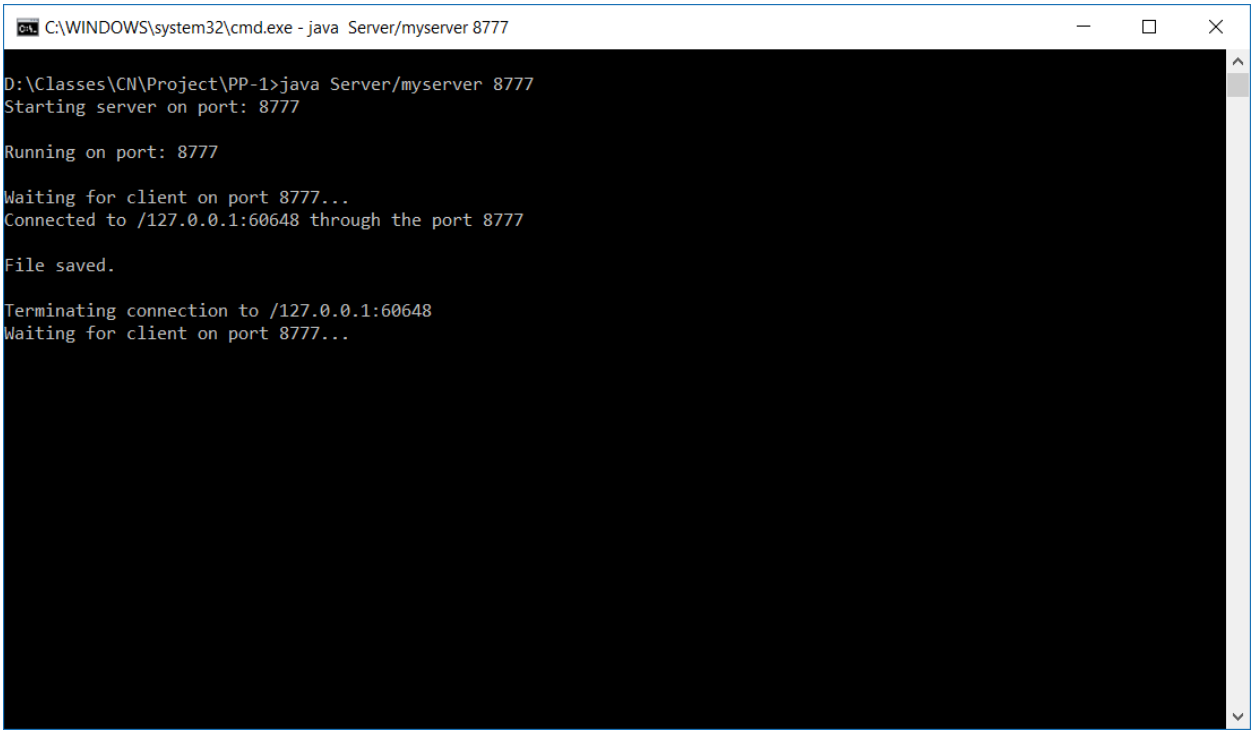
Execution - java Server/myserver 8777



LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

ITCS 6166 – COMPUTER NETWORKS – SPRING 2016 – Programming Project 1





LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

Part-2 TCP vs. UDP

Steps to Execute:

For TCP Readings	For UDP Readings
First Execute the tcp_receiver javac tcp_receiver.java java tcp_receiver **Then execute tcp_sender** javac tcp_sender.java java tcp_sender	**First Execute the udp_receiver** javac udp_receiver.java java udp_receiver **Then execute upd_sender** javac udp_sender.java java udp_sender

TCP Readings:	UDP Readings:
Step 1 - 10 Characters Maximum ETE: 146.0 Average Time: 1.56 Max ETE at: 984 Zero Delay: 775	**Step 1 - 10 Characters** Maximum ETE: 141.0 Average Time: 2.723 Max ETE at: 363 Zero Delay: 700
Step 2 - 10 Characters 5 Times Maximum ETE: 137.0 Average Time: 1.517 Max ETE at: 787 Zero Delay: 755 Maximum ETE: 9.0 Average Time: 1.152 Max ETE at: 118 Zero Delay: 762 Maximum ETE: 9.0 Average Time: 1.236 Max ETE at: 570 Zero Delay: 737 Maximum ETE: 403.0 Average Time: 3.578 Max ETE at: 686 Zero Delay: 760 Maximum ETE: 15.0 Average Time: 1.076 Max ETE at: 163 Zero Delay: 773	**Step 2 - 10 Characters 5 Times** Maximum ETE: 9.0 Average Time: 1.222 Max ETE at: 236 Zero Delay: 764 Maximum ETE: 9.0 Average Time: 1.121 Max ETE at: 193 Zero Delay: 776 Maximum ETE: 16.0 Average Time: 1.232 Max ETE at: 926 Zero Delay: 768 Maximum ETE: 9.0 Average Time: 1.134 Max ETE at: 260 Zero Delay: 765 Maximum ETE: 9.0 Average Time: 1.329 Max ETE at: 122 Zero Delay: 744

LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

Step 3 - 200 Characters 5 Times	**Step 3 - 200 Characters 5 Times**
Maximum ETE: 15.0 Average Time: 1.076 Max ETE at: 163 Zero Delay: 736	Maximum ETE: 323.0 Average Time: 1.819 Max ETE at: 338 Zero Delay: 785
Maximum ETE: 11.0 Average Time: 1.305 Max ETE at: 150 Zero Delay: 736	Maximum ETE: 9.0 Average Time: 1.259 Max ETE at: 1 Zero Delay: 757
Maximum ETE: 11.0 Average Time: 1.477 Max ETE at: 353 Zero Delay: 745	Maximum ETE: 9.0 Average Time: 1.36 Max ETE at: 67 Zero Delay: 733
Maximum ETE: 146.0 Average Time: 1.903 Max ETE at: 95 Zero Delay: 754	Maximum ETE: 9.0 Average Time: 1.164 Max ETE at: 245 Zero Delay: 763
Maximum ETE: 149.0 Average Time: 1.847 Max ETE at: 831 Zero Delay: 709	Maximum ETE: 143.0 Average Time: 1.095 Max ETE at: 96 Zero Delay: 752
Step 4 - 1000 Characters 5 Times	**Step 4 - 1000 Characters 5 Times**
Maximum ETE: 594.0 Average Time: 3.508 Max ETE at: 87 Zero Delay: 656	Maximum ETE: 144.0 Average Time: 3.321 Max ETE at: 430 Zero Delay: 693
Maximum ETE: 123.0 Average Time: 2.91 Max ETE at: 835 Zero Delay: 603	Maximum ETE: 138.0 Average Time: 2.67 Max ETE at: 693 Zero Delay: 648
Maximum ETE: 89.0 Average Time: 2.58 Max ETE at: 907 Zero Delay: 670	Maximum ETE: 66.0 Average Time: 2.217 Max ETE at: 57 Zero Delay: 667
Maximum ETE: 147.0 Average Time: 3.335 Max ETE at: 670 Zero Delay: 616	Maximum ETE: 74.0 Average Time: 2.017 Max ETE at: 420 Zero Delay: 667
Maximum ETE: 125.0 Average Time: 2.328 Max ETE at: 417 Zero Delay: 678	Maximum ETE: 140.0 Average Time: 1.979 Max ETE at: 632 Zero Delay: 701

LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

Answer 1:

The average ETE values do not vary much as seen.

This is because the process we are repeating 5 times is using the same network connection (LAN) but the small variations are due to the utilizations of the network.

Answer 2:

The Average ETE values have increased for steps 2, 3, 4. This is because of the number of segments being transferred has increased for every step as the string length is increasing.

The Maximum ETE has increased for steps 2, 3, 4. Abrupt changes can be observed as there may be some packet losses or retransmissions involves in TCP.

Answer 3:

Step 2:
The average ETE and the maximum ETE values for TCP are greater than the values of the UDP. This is because TCP sends acknowledgement for the messages received so the higher values.

Step 3:
The average ETE and the maximum ETE values for TCP are greater than the values of the UDP. This is because of the increasing message size from step 2 and as a result more segments would be needed to transmit the message.

Step 4:
The average ETE and the maximum ETE values for TCP is much greater than the values of the UDP. This is because the padding is 1000 in this case so more segments for TCP.

Differences between TCP and UDP:

- TCP is a connection oriented whereas UDP is Connectionless as a result TCP takes more time to transfer the data. The same can be seen from the data collected.
- TCP is used for the applications for high reliability and for with the transmission time is if less concern on the other hand applications use UDP for fast transmission as it dumps the data at the receiver
- UDP is faster than the TCP as observed from the data collected.

Performance:

From these observations it is evident that the UDP performance is better in case of packet transmission rate and the number of zero delay values are greater in UDP case leading to the low overall average delay. But in case of UDP the there is no connection setup and packets can be dropped on the way to the recipient

LAXMAN SAI KARTHIK MAHIDHAR GOPALABHATLA
800936333
LGOPALAB@UNCC.EDU

Answer 4:

If the tests are run over the Internet where there is a high chance of a packet loss, the Average ETE values for TCP would be larger than UDP as TCP is a reliable service. As a result more delay will be caused due to packet loss for retransmission of lost packets

In case of UDP the lost packet would be discarded and would not be included for ETE as it can tolerate the loss unlike TCP.