Lab 1.2 (Mandatory): Iterative server questionaire

Started: 28 Sep at 16:06

Quiz instructions

You will only get a single attempt at this quiz. Fill your answers in the appropriate areas.

Submit this quizz only once per group!

Question 1

0 pts

Give your group number

5

Question 2

1 pts

The listening socket is bound to a specific address. What address is this? (Give both the symbolic name used in the code, and the corresponding IPv4 address in numeric or dotted notation).

HTML Editor

12pt Paragraph

The symbolic name is INADDR_ANY

The IP address is 127.0.0.1

p 10 words

Question 3

1 pts

In the code, there is a call to **recv**() as follows:

```
ret = recv( cd.sock, cd.buffer, kTransferBufferSize, 0 );
```

The return value ret will be one of the following:

1. ret = −1
2. ret = 0
3. 0 < ret < kTransferBufferSize
4. ret = kTransferBufferSize

Describe the implications of each case!

Also answer why cd.buffer (see ConnectionData declaration) is defined to be of size kTransferBufferSize**+1** rather than just plain kTransferBufferSize.

HTML Editor

12pt Paragraph

1. ret = -1

This means that there is an error in recv()

2. ret= 0

This means that the shutdown process was orderly done

3.   0 < ret < kTransferBufferSize

This means that the data received in the buffer, measured in bytes, is between the range of between 0 and 64 bytes

4. ret = kTransferBufferSize

This means that 64bytes were actually received based on the recv() function

Since kTransferBufferSize starts from 0, 1is added to make space for track the end of

p 114 words

Question 4

1 pts

Sending is performed using the **send**() method as follows:

```
ret = send( cd.sock,
  cd.buffer+cd.bufferOffset,
  cd.bufferSize-cd.bufferOffset,
  MSG_NOSIGNAL
);
```

How does the **send()** method indicate that the connection in question has been closed/reset?
How does **MSG_NOSIGNAL** relate to this (on linux machines)?

HTML Editor

12pt Paragraph

The send() method is used to empty the buffer and change the connection state to a new receiving state

MSG_NOSIGNAL: This does not allow for Linux kernel to generate SIGPIPE signal if the peer on a stream-oriented socket has closed or reset the connection. SIGPIPE is a signal sent to a process when it attempts to write a pipe without a process connected to the other end

p 67 words

Question 5

1 pts

Discuss the reasons for this behaviour with your partner. Why are these two strategies used?

Also, quickly look through the error codes (values of `errno`) possible after `accept()`, `send()`, and `recv()` (check the *man*-pages!). Under which conditions attempting to continue executionmight be unreasonable?

HTML Editor

12pt Paragraph

The first approach is an exit method based on the status of the connection. If it is unable to accept or bind, or create a connection, then it simply exits.

The second approach, is to drop requests from clients that cannot be established and start a new process to connect newer connections from another client, that maybe in waiting. Since it is an iterative server, it processes on connection at a time.

p

72 words

Question 6

1 pts

Discuss with your partner: How is the program notified that a connectionattempt has failed or succeeded?

Hint: the process is described in the course book!

HTML Editor

12pt

Paragraph

The program decides on what to do based on the connection errors. If the number of errors, as coded, is greater than zero, it goes ahead to compare this with the number of clients. Then, if the number of connections is equal to the number of clients, then it returns that all connection are in error and has failed.

If the above does not hold, then it returns the total number of successful connections on the server. Even if one client connection fails, it will return error for this client but the server will continue with the other connections.

p 99 words

Question 7

1 pts

Try to send messages with each of the clients. Describe the results – do you receive a response immediately?

Check with *netstat* and document the status of the connection from each client.

HTML Editor

12pt Paragraph

Once we had established connections to different clients, we sent messages from the first and the second. The server responded to immediately only the first one and seem to discard the messages from the second. We also got a message "response does not match original query" a times.

The netstat command shows the connection status of the sockets and their corresponding process identifiers (PID)

p 64 words

Question 8

1 pts

When you disconnected the first client, what happened? Explain why.

HTML Editor

12pt Paragraph

After an orderly shutdown of the connection with the first cl established for the second client, and the messages being sent from this client now started getting responses from the server. We take it that the iterative server was only responding to one client at a time. When we ran the netstat command, the PID for the first client was nom longer seen on the list. We can infer that this is simply how the iterative server works

p

81 words

Question 9

1 pts

Measure the round trip time when the client and server are running on the same machine. Also measure the round trip time when they are on different machines.

Can you observe any differences? Write down the times. (Note: take the average of a few (> 5) attempts.)

HTML Editor

12pt Paragraph

Size of Data Sent (Bytes)	RTT (ms)	RTT (ms)
	Client & Server are on the same machine	Client & Server on Different Machine
10	0.108161	10.770205
26	0.100951	9.161370
27	0.061767	9.626789
30	0.084463	5.936388
33	0.092891	11.162449
54	0.083655	4.706442
104	0.061767	9.626789
Average value	0.08480786	8.71291886

Yes, we observed clear differences between the scenarios. The response was far better, lower in value, when the server and client were running on the same machine. The response time was higher when the client connected remotely to the server.

table » tbody » tr » td 86 words

Question 10 1 pts

Measure the round trip times for two concurrently connected simple clients (similar to exercise I.c.1).

Discuss with your partner: What is the largest factor in the measured round trip time of the second client?

HTML Editor

12pt Paragraph

To handle concurrent connections the iterative server program has to be modified to accept simultaneous connections from clients. The iterative server takes one connection at a time, and so we could not initiate two concurrent sessions.

Intuitively, for concurrent sessions, the server response time will play a major role in creating some difference in the round trip time. The number of hops, distance, traffic level and transmission channel/medium will be the same for both clients at the same time

p 80 words

Question 11

1 pts

Run the above command (make sure that the server is still running), and note the results.

Increase the number of clients a few times (try, for example, using 10, 15, 30, 50 and 100 clients). What happens to the minimal/maximal times required to establish a connection? What happens to the round-trip times? Did any errors occur?

HTML Editor

12pt Paragraph

We observed that as the number of clients was being increased, the minimal and maximal times, also increased as well.

Also, the RTT values increased as the number of clients increase.

We therefore see a direct proportionality between the number of clients and the minimal/maximal/RTT values.

Yes, we observed a few errors that seem to also increase with the number of clients. For example, we got:

conn 23 : error in recv() : Connection reset by peer

conn 25 : error in recv() : Connection reset by peer

p 86 words

Question 12

1 pts

How long did it take for the connection attempts to time out?

HTML Editor

12pt Paragraph

We started our tests for this session at 15:18:50 (28-09-2020) and until 17:48:55, we still have not gotten anything.

Based on our group discussion for denial-of -service, we were expecting for the server to refuse taking newer queries, connections and just freeze on processing generally.

p

49 words

Saved at 17:51

Submit quiz