# Quiz 2: Client-Server Computing

**Due** Jun 1 at 11:59pm **Points** 100 **Questions** 5
**Available** May 26 at 8am - Jun 1 at 11:59pm 7 days **Time Limit** 60 Minutes

This quiz was locked Jun 1 at 11:59pm.

## Attempt History

| | Attempt | Time | Score |
|---|---|---|---|
| **LATEST** | Attempt 1 | 55 minutes | 69.17 out of 100 |

Score for this quiz: **69.17** out of 100
Submitted Jun 1 at 4pm
This attempt took 55 minutes.

### Question 1

**15 / 15 pts**

What is the name of the class in Java for creating server-side TCP sockets? ServerSocket

What is the name of the class in Java for creating client-side TCP sockets? Socket

What is the name of the class in Java for creating UDP sockets? DatagramSocket

**Answer 1:**

Correct!

ServerSocket

**Answer 2:**

Correct!

Socket

**Answer 3:**

Correct!

DatagramSocket

## Question 2

6.67 / 20 pts

In the FTP protocol, the client accepts data connection requests from the server when data is transferred in [active] mode, whereas the server accepts data connection requests from the client when data is transferred in [passive] mode. Passive mode should be used when the [server] is behind a firewall/NAT, since the [client] reply includes a public IP address to be connected to. Active mode should be used when the [client] is behind a firewall/NAT, since the [server] command includes a public IP address.

**Answer 1:**

Correct!

active

**Answer 2:**

Correct!

passive

**Answer 3:**

ou Answered

server

orrect Answer

client

**Answer 4:**

ou Answered client

orrect Answer PASV

**Answer 5:**

ou Answered client

orrect Answer server

**Answer 6:**

ou Answered server

orrect Answer PORT

## Question 3

22.5 / 25 pts

Since an RPC request message may be lost, the client must use retransmission to make sure the server eventually gets the request. The problem is that the server may get multiple requests. Therefore a message must be tagged with a sequence number to detect this. This requires that the client and server share state, complicating the semantics when the server may crash. The three kinds of semantics for RPC with server crashes are at least once semantics (client keeps trying), at most once semantics (client gives up), and exactly once semantics. The latter may be difficult or impossible to attain.

**Answer 1:**

ou Answered multiple

orrect Answer duplicate

**Answer 2:**

Correct! sequence number

**Answer 3:**

Correct! at least once

**Answer 4:**

Correct! at most once

**Answer 5:**

Correct! exactly once

## Question 4

10 / 20 pts

Match the communication mechanism below with the corresponding explanation.

Correct! **LRPC** switches client thread dir

ou Answered **Fbufs** copies data between buf

Correct Answer shares buffer memory space between layers in the network stack

ou Answered **Network communication stack** shares buffer memory sp

| | | |
|---|---|---|
| | **Correct Answer** | copies data between buffers in network service layers |
| Correct! | **RPC on same machine** | context-switches througl |

## Question 5

**15 / 20 pts**

Suppose a print server writes a log record to disk every time it performs a request, to avoid re-executing repeat requests from clients if it crashes and reboots. Every print request is checked with the log, to see if that request has already been performed.

1. Suppose the record is written **before** performing the print request. Why does this not ensure exactly-once semantics?
2. Suppose the record is written **after** performing the print request. Why does this not ensure exactly-once semantics?

Your Answer:

1. There is no confirmation that the request was completed. There could have been a crash after the record was written but before the print request.
2. If the print request was made and then a crash occurs before the record is written, the request log is lost.

1. The server may crash after writing the log record but before performing the request. Then the request may never be performed.
2. The server may crash after performing the request but before writing the log record. Then the request may be executed again if the client retransmits.

Quiz Score: **69.17** out of 100