Lebanese University
Faculty of Sciences
-Branch 1 -

الجامعة اللبنانية
كلية العلوم
-الفرع الأول-

Master 1 - Semester 2
University Year 2024 - 2025
Final Exam
Info 408- Distributed System
Unauthorized documents

June 2025 | **Grading scale / 100** | **Duration 90 min**

Exercise 1:

Write a Java program using UDP sockets where multiple clients can send a string and a command to the server. The server performs the requested operation on the string and returns the result.

Server side:

a. Listens for incoming UDP packets on a specified port.
b. Receives a message (COMMAND and message)
   where COMMAND can be:
   - UPPER: Converts the message to uppercase
   - REVERSE: Reverses the message
   - LENGTH: Returns the number of characters
c. Sends the result of the operation back to the client.

Client side:

a. Prompts the user to input a string and choose an operation (UPPER, REVERSE, LENGTH).
b. Sends the message (with the COMMAND and the message) to the server using UDP.
c. Receives the result from the server.
d. Displays the result to the user.

Exercise 2:

Create a client-server application using TCP sockets in Java where multiple clients can connect to a central server to perform authenticated operations.

Client side:

1. The client connects to the server using the server's IP address and port number.
2. Upon connection, the client is prompted to enter a username and password.
3. If authentication succeeds (based on a predefined user list on the server), the client can issue the following commands:
   a. ADD x y → Server returns the sum of x and y.
   b. MAX x y → Server returns the maximum of x and y.
   c. BYE → Ends the session.
4. The client receives and displays the server's response for each command.
5. Invalid commands should return an error message from the server.

Server side:

1. Maintains a predefined list of valid usernames and passwords.
2. Accepts multiple client connections concurrently using threads.
3. Authenticates each client upon connection. If authentication fails, the server sends a rejection message and closes the connection.
4. Upon successful login, the server handles commands from the client as described above.
5. Closes the connection when the client sends BYE.

Exercise 3:

Write a Java program that uses RMI (Remote Method Invocation) to count vowels and consonants in a given string.

Server Side:

1. Define a remote interface named TextAnalysisService with methods:
   - int countVowels (String input)
   - int countConsonants (String input)
2. Implement this interface in a class called TextAnalysisServiceImpl, where:
   - countVowels () returns the number of vowels (a, e, i, o, u)
   - countConsonants () returns the number of consonants
3. Host the remote object using RMI registry.

Client Side:

1. Develop a client that connects to the remote TextAnalysisService.
2. Prompt the user to input a sentence.
3. Call the remote methods to get the number of vowels and consonants.
4. Display the results to the user.