Q1. Briefly describe five challenges of distributed systems.

Q2. Discuss any two architectural models for the construction of distributed systems. Explain a suitable application scenario for each.

Q3. What are the advantages of using multiple threads over multiple processes? Explain the differences between a worker pool thread model and a hread-per-object model.

Q4. In the context of Remote Invocation in the distributed system, provide definitions for the following terms:
(a) (1 mark) At-most-once invocation semantics
(b) (1 mark) At-least-once invocation semantics
(c) (1 mark) Remote interface
(d) (1 mark) Stub
(e) (1 mark) Registry

Q5. Discuss the architecture of a microkernel-based operating system. Comment on how well this architectural model supports the creation of extensible operating systems.

Q6. What is the purpose of the Flat File Service in the File Service Architecture (FSA)? Discuss the differences between the Flat File Service Interface and the Unix File Interface.

Q7. In the context of designing a distributed tic-tac-toe gaming system, suppose the game states between clients and the server must be exchanged via an indirect communication mechanism. Identify the most appropriate indirect communication approach for this purpose and provide the rationale for your choice.

Q8. With the help of a diagram, explain how recursive server-controlled navigation works. In which scenario DNS should use recursive server-controlled navigation?

Q9. Draw a diagram and explain how two network entities (e.g., Alice and Bob) can exchange a shared secret key with the help of an authentication server.

Q10. What is a digital signature? What is the role of the certificate chain in authenticating an individual’s identity?

Q11. Write a simple Java RMI program that demonstrates the invocation of remote object services. Implement a service that offers dictionary services. It should support 3 operations/services

(a) "ADD" which adds a new word and its meaning to the dictionary;

(b) "SEARCH" which returns the meaning of a word passed as an argument.

(c) DELETE which deletes the word passed as an argument from the dictionary.

Write both server and client programs.

Requirements:

- Binding RMI object to the registry.
- Defining interfaces for various services (e.g., ADD, SEARCH, DELETE).
- Implementation of RMI interfaces.
- Looking up the remote object and invoking remote methods.
- Handling of two distinct types of errors or exceptions in the system.