1. Briefly explain these terminologies. If they are acronyms, also write what they stand for. (15 points)

(a) COW - Clusters of Workstations (COWs) are a collection of standard PCs or workstations connected through communication components.

(b) DSM - Distributed Shared Memory (DSM) is a large virtual address space built from the virtual memory of each individual node.

(c) RSVP - Resource reSerVation Protocol (RSVP) is a transport-level control protocol for enabling resource reservations in network routers.

(d) Object Adapter - It is a mechanism to group objects per policy and acts a wrapper around the implementation of an object.

(e) ACL - Agent Communication Language (ACL) is an application level protocol for agents to communicate.

2. Briefly describe DOS, NOS, and middleware and the main goal for each system mentioned above. (12 points)

<table>
<thead>
<tr>
<th>System</th>
<th>Description</th>
<th>Main goal</th>
</tr>
</thead>
<tbody>
<tr>
<td>DOS</td>
<td>Tightly-coupled operating system for multiprocessors and homogeneous multicomputers</td>
<td>Hide and manage hardware resources</td>
</tr>
<tr>
<td>NOS</td>
<td>Loosely-coupled operating system for heterogeneous multicomputers (LAN and WAN)</td>
<td>Offer local services to remote clients</td>
</tr>
<tr>
<td>Middleware</td>
<td>Additional layer atop of NOS implementing general-purpose services</td>
<td>Provide distribution transparency</td>
</tr>
</tbody>
</table>

3. What is a microkernel? Describe the benefits using microkernels. (8 points)

(a) A microkernel is the core part of the operating system containing only a minimal set of services allowing traditional operating system services to be provided by special server processes executing in user space.

(b) Using microkernels has following benefits:
   i. **Modularity**: The services of the operating system can be divided into different user-level modules.
   ii. **Flexibility**: It is easy to replace a module.
   iii. **Stability**: Modules can be placed on different machines.
   iv. **Reliability**: A service fault does not bring down the system.
   v. **Protection**: This separation of kernel and server allows protection within the operating system itself.
   vi. **Heterogeneity**: It allows different operating systems to coexist.

4. Describe a general pattern followed by a client and server for connection-oriented communication using socket functions in UNIX. (8 points)

```plaintext
  +-----------+| socket() |+-----+-----+|V well-known port | bind() |
              +-----+-----+|V +-----------+ +-----------+| socket() | | listen() |+-----+-----+ Synchronization Point +-----+-----+| connect() |<-------------------------------->| accept() |
              +-----+-----+ +-----+-----+ +-----------+ data (request) +-----------+ | write() |--------------------------------->| read() |<--+
              | | | +-----+-----+ +-----+-----+ | v v | v v v | +-----------+ data (reply) +-----------+ | v | write() | <---
              +---| write() |---------------------------------| write() |---++-----+-----+ +-----+-----+| v v v +-----------+ +-----------+| close() | | close() |+-----------+ +-----------+```

5. Consider the following code:

```c
int f(x,y)
{
    x = x - 1;
    y = x * y - 1;
    return(x + y);
}

main()
{
    int a, b;
    a = 3;
    b = f(a, a);
    printf("a = %d, b = %d \n", a, b);
}
```

What will be printed if the parameters are passed using:

(a) call-by-value?
(b) call-by-reference?
(c) call-by-copy/restore using Pascal semantics?

Which of the above parameter passing method(s) cannot be easily implemented using remote procedure calls? Explain. (8 points)

(a) i. a = 3, b = 7
ii. a = 3, b = 6
iii. a = 2, b = 7 (Pascal semantics)

(b) Call-by-reference cannot be easily implemented because the reference addresses are different from the local and remote machines.

6. (a) What is an idempotent operation? (3 points)
(b) Discuss whether the following operations are idempotent? (6 points)
   - Write a block of data into a file
   - Request the number of users currently logged into a remote system
   - Deposit money into a bank account
(c) In which invocation semantics of RPC is an idempotent operation required? (3 points)

(a) An idempotent operation is an operation that can be performed repeatedly with the same effect.
(b) • Writing a block of data into a file is an idempotent operation because the result file is same.
   • Requesting the number of users currently logged into a remote system is an idempotent operation because each request does not change the number of users in the remote machine.
   • Depositing money into a bank account is not an idempotent operation because each deposit could add the account balance.
(c) In at-least-once invocation semantics an idempotent operation is required.
7. (a) What is persistent communication? In which case(s) do we use persistent communication? (4 points)
(b) What is a message broker? What is its purpose? (4 points)

(a) i. Persistent communication is a type of communications in which a message is stored at a communication server as long as it takes to deliver it at the receiver.
ii. Persistent communication can be used in the applications in which the sending or receiving applications need not be executing during the transmission of messages such as e-mail or network news.

(b) A message broker is a special node that converts messages in heterogeneous message-queuing systems. Its purposes are:
   i. It can transform incoming messages to target format, possibly using intermediate representation
   ii. It may provide subject-based routing capabilities
   iii. It can act very much like an application gateway.

8. What is a data stream? Describe three different transmission modes of data streams. (8 points)

(a) A data stream is a sequence of data units. It applies to both discrete and continuous stream.
(b) i. **Asynchronous** data stream has no restrictions with respect to when data is to be delivered.
ii. **Synchronous** data stream is required to define a maximum end-to-end delay for individual data packets.
iii. **Isochronous** data stream is required to define maximum and minimum end-to-end delay jitter is bounded.

9. Describe the following four types of servers: iterative server, concurrent server, stateless server, and stateful server. (12 points)

(a) A iterative server itself handles the request itself and can handle only one client at a time.
(b) A concurrent server does not handle the request itself but pass it to a separate thread or another process.
(c) A stateless server never keeps accurate information about the status of a client after having handled a request.
(d) A stateful server keeps track of the status of its clients.

10. Describe reasons for migrating code and models for code migration. (12 points)

(a) Reasons for migrating code:
   - Overall system performance can be improved
   - Code migration makes sense to process data close to where those data reside.
   - Code migration exploits parallelism.
   - Dynamically configuring distributed systems increases flexibility.

(b) **Weak mobility**: Only code and initialization data segment are moved.
**Strong mobility**: Segments with execution state are also moved.
**Sender-initiated**: The migration is initiated at the machine where the code currently resides or is being executed.
**Receiver-initiated**: The initiative for code migration is taken by the target machine.

11. What is a software agent? Describe three different types of agents. (10 points)

(a) A software agent is an autonomous process capable of reacting to, and initiating changes in, its environment, possibly in collaboration with users and other agents.
(b) i. **Collaborative agent** can collaborate with others in a multiagent system.
   ii. **Mobile agent** can move between machines.
   iii. **Interface agent** can assist users in the use of one or more applications (usually with learning ability).
   iv. **information agent** can manage information from physically different sources.
12. Consider an application that implements a remote dictionary. The dictionary offers four operations: `count` to return the word count, `insert` to insert a word and its content, `delete` to delete a word, and `lookup` to search for a word and return the content of the word. (12 points)

(a) Use Sun RPC to specify this application.
(b) Specify this application in a Java RMI interface file.

(a) Sun RPC

```
struct word_content {
    string word<>
    string content<>
};

program RDICTPROG {
    version RDICTVERS {
        int COUNT(void) = 1;
        int INSERT(word_content) = 2;
        int DELETE(string) = 3;
        string LOOKUP(string) = 4;
    } = 1;
} = 0x30090949;
```

(b) Java RMI interface file

```
import java.rmi.*;
public interface DictionaryInterface extends Remote {
    public long count() throws RemoteException;
    public long insert(String word, String content) throws RemoteException;
    public long delete(String word) throws RemoteException;
    public String lookup(String word) throws RemoteException;
}
```