



**UNIVERSITY OF COLOMBO, SRI LANKA**

**UNIVERSITY OF COLOMBO SCHOOL OF COMPUTING**

**DEGREE OF BACHELOR OF INFORMATION TECHNOLOGY (EXTERNAL)**

**Academic Year 2019 – 3<sup>rd</sup> Year Examination – Semester 6**

***IT6505: Middleware Architecture***

**09<sup>th</sup> November, 2019**

**(TWO HOURS)**

**To be completed by the candidate**

BIT Examination Index No: .....

**Important Instructions:**

- The duration of the paper is **2 (two) hours**.
- The medium of instruction and questions is English.
- Part A of paper has **4 questions** and **14 pages**.
- **Answer all questions.** All questions **carry** equal marks.
- **Write your answers** in English using the space provided **in this question paper**.
- Do not tear off any part of this answer book.
- Under no circumstances may this book, used or unused, be removed from the Examination Hall by a candidate.
- Note that questions appear on both sides of the paper.  
If a page is not printed, please inform the supervisor immediately.
- Calculators are **not** allowed.

**Questions Answered**

Indicate by a cross (×), (e.g. ☐ **×** ) the numbers of the questions answered.

To be completed by the candidate by marking a cross (×).	Question numbers				
	1	2	3	4	
To be completed by the examiners:					

1)

- (a) Global tech giants such as Google, Amazon, Apple and Facebook use service and data distribution as a means of guaranteeing an overall reliable service. Reliability is a key aspect of a distributed system.

- (i) Briefly explain what is meant by reliability in a distributed system.
- (ii) Using an example each, explain three (03) types of reliability concepts expected in a distributed system.

(8 marks)

**ANSWER IN THIS BOX**

- (i) Reliability denotes the ability of a distributed system to deliver its services even when one or several of its software or hardware components fail. → **2 marks**

(ii)

**Fault tolerance** is the ability of a distributed computing system to recover from the failure of some component. A component is considered faulty once its behavior is no longer consistent with its specification. *Eg. Imagine a cloud storage, if one server is faulty another should make the data available for the client.*

**High availability** is a system provides uninterrupted service in spite of failures

**Consistency** is the ability of a distributed computing system to coordinate failures. It behaves like a non-distributed system.

**Security** is the ability of a system to protect data, services and resources against unauthorized access

**Privacy** is the ability of a system to protect user identity and data from other users

→ **3 x 2 marks (1 mark for description and 1 mark for example)**

- (b) Define the concepts of fault, error and failure in relation to a system. Explain using examples four (04) types of failures in a distributed system.

(6 marks)

**ANSWER IN THIS BOX**

**FAILURE:** A failure is the inability of a software system or component to perform its required functions within specified performance requirements. →

**2 marks**

Crash Failures: Occur across the server of a typical distributed system and when these occur the operation of the server halts. Example is when the operating system of the server fails.
Timing Failures: Occur when the server response time is higher than the expected response time. Example when a server is busy it may timeout the client request.
Omission Failures: Occurs when there is a lack of a reply or response from the server. Example when there is a buffer overflow or server is not listening.
Network Failures: When the network fails.
Byzantine failures: Occur when there is an arbitrary failure in a system. Example when there is a virus infection etc.
→ <b>4 x 1 mark</b>

- (c) Consider *gpacompute.x* which is an RPC IDL file. The file defines a data structure named marks that consists of five subject marks (floating point) and a function named GPA which calculates the grade-point-average (gpa) for the marks using the data structure and returns the gpa value (floating point).

- (i) What is used to compile the RPC IDL file?
- (ii) When compiled, what are the generated files and what purpose do each of the generated files serve?

**(4 marks)**

**ANSWER IN THIS BOX**

- (i) rpcgen is the programme used → **1 marks**

- (ii)
  - gpacompute.h – a header file used to define the contracted function
  - gpacompute\_clnt.c – client stub which marshals and unmarshals the parameters
  - gpacompute\_svc.c – server skeleton which marshals and unmarshals the parameters

→ **3 x 1 mark**

(iii) Write down the code for the gpacompute.x RPC IDL file and explain each line of code.

(7 marks)

**ANSWER IN THIS BOX**

```
#define VERSION_NUMBER 1

struct marks { -> 1 mark
    float m1; -> 1/2 mark
    float m2;
    float m3;
    float m4;
    float m5;
}; /* Define the marks data structure*/ -> 1/2 mark

program GPACOMPUTE_PROG { /* Define the gpacompute program*/ -> 1
mark
    version GPACOMPUTE_VERSION {
        float GPA(marks) = 1; /* Define the add function */ -> 2
mark
    } = VERSION_NUMBER; /* Assigns the version number */ -> 1 mark
} = 1111111111; /* Define the identifier*/ -> 1 mark
```

2)

- (a) Explain in detail what is a Message Queue is, and the role of Message Oriented Middleware.

(4 marks)

**ANSWER IN THIS BOX**

A message queue is like a fast mailbox, where a message can be put even with or without the recipient being online. To put a message to the queue, the programme does a Put; and to take a message from the queue, the programme does a Get. Message oriented middleware is a Programme-to-Message Queue middleware. → **2 mark**

The middleware does the transferring of messages from one queue to another, and ensures that what ever happens in the network the message arrives eventually and that only one copy of the message is placed in the destination queue. → **2 mark**

- (b) A message broker can be considered an intermediary programme module that facilitates process interaction. Clearly explain the functionality of a message broker and, using a real world example the need for message brokers.

(5 marks)

**ANSWER IN THIS BOX**

A message broker translates a message from the formal messaging protocol of the sender to the formal messaging protocol of the receiver. Message brokers are elements in telecommunication or computer networks where software applications communicate by exchanging formally-defined messages. → **2 mark**

A message broker is an architectural pattern for message validation, transformation, and routing. It mediates communication among applications, minimizing the mutual awareness that applications should have of each other in order to be able to exchange messages, effectively implementing decoupling.

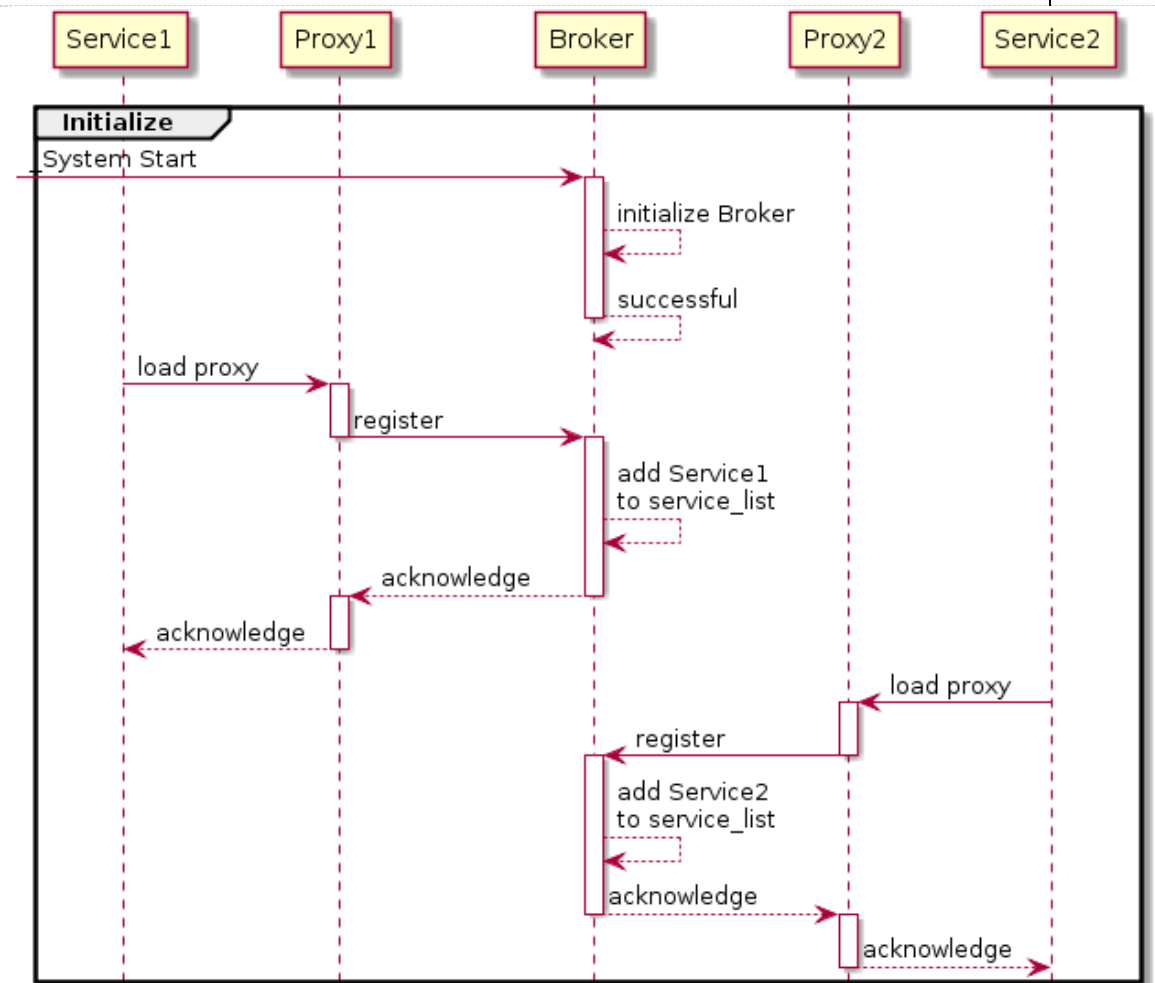
For a real world example → **3 mark**

- (c) Suppose there are two parties named *service1* and *service2*, where each party has a proxy of its own called *proxy1* and *proxy2* respectively. Between the parties there exists a message broker. Service1 requires service from service2 and is required to use the service broker to communicate to the other party.

- (i) Using an interaction diagram illustrate how the overall system would initialize before any party is able to communicate with each other.

(5 marks)

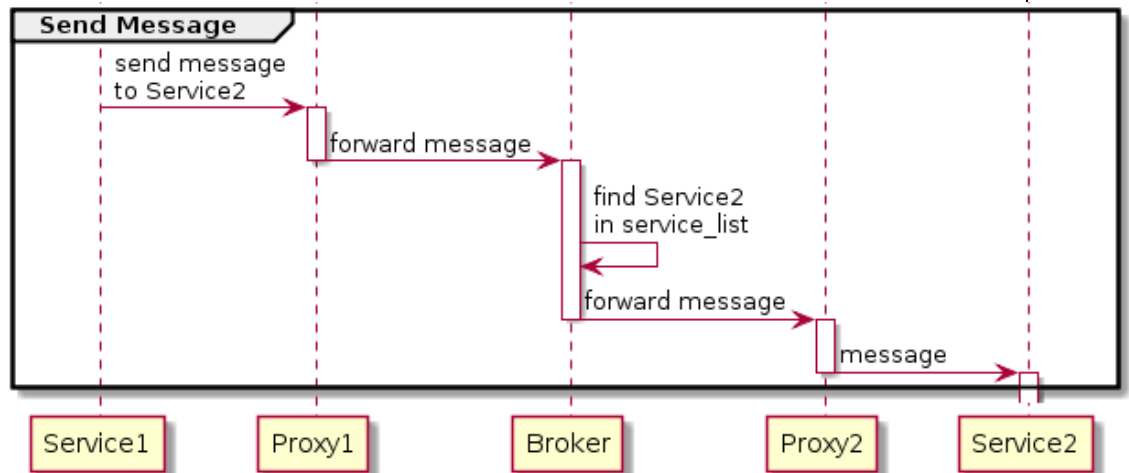
**ANSWER IN THIS BOX**



- (ii) Using a diagram illustrate how the system would handle a request from service1 for a service offered by Service2.

(3 marks)

**ANSWER IN THIS BOX**



- (d) Message Oriented Middleware (MOM) may behave synchronously or asynchronously. Using a real world example clearly explain an advantage of this property of MOM.

(4 marks)

**ANSWER IN THIS BOX**

Assume a system that may be offline at different times → **2 mark**, or may be communicating via unreliable networks → **2 mark**, the message queues would hold the messages until the systems are operational or the networks can complete the communication. This would still allow for services to be offered to service requesters.

- (e) A database transaction is required to conform to the well known ACID properties. A system is required to perform a distributed database transaction. The distributed transaction is required to update multiple databases on different subsystems.

Assuming that one or more of the databases fail to update, clearly state how a MOM would rectify the problem.

(4 marks)

**ANSWER IN THIS BOX**

If one or more databases fail to update, then the updated databases (sub-transactions) must be rolled back. That is when a subsystem fails to update its own database then it should inform other systems by "Putting" a message to the other subsystems to rollback their relevant transaction if they have already committed it in their databases. Committing is irrevocable, and if one has aborted, the whole system would abort. Any rebooted system would read stable log and go along with final result?

3)

- (a) What is meant by Object Serialization in a typical client server interaction? List the usages of Object Serialization.

(5 marks)

**ANSWER IN THIS BOX**

Serialization is the process of converting an object (state of an object) into a stream of bytes.

→ 2 marks

- Save the state of an object in order to be able to recreate it when needed. (E.g. save objects in memory, as files, JSON structures, or in databases)
- Communicate objects through a network. A serialized object can be transmitted from one location to another location and after receiving the byte stream can be de-serialized to reproduce the same object.

→ 1.5 mark \* 2



- (b) Explain The **functionality of the code** given below, while describing **keywords in bold**.  
(5 marks)

```
@Entity
@Table(name = "tbl_comment")

public class Comment implements Serializable{
    @Column(name="comment", length = 1024)
    private String comment;
    ...

    public String getComment() {
        return comment;
    }

    public void setComment(String comment) {
        this.comment = comment;
    }

    ...

    @Override
    public String toString() {
        JSONObject jsonObject = new JSONObject(this);
        return jsonObject.toString();
    }
    ...
}
```

### **ANSWER IN THIS BOX**

The above code can be seen as a Model class written for a Java middleware application. → **1 marks**

Purpose of this class is to bind table columns of tbl\_comment with instance variables of the class (e.g. instance variable 'private String comment' is mapped to the column 'comment'). → **2 marks**

Getter and Setter methods are used to read and modify state of the object. → **1 marks**

'toString()' method handles the serialization process (which returns a JSON object). This will enable the persisting and communicating capabilities for Comment objects. → **1 marks**

(c) List **four** differences between **SOAP** and **Restful** Web services.

(6 marks)

**ANSWER IN THIS BOX**

- REST is an architectural style. SOAP is a protocol.
- SOAP uses services interfaces to expose the business logic while REST uses URI to expose business logic.
- SOAP defines standards to be strictly followed. REST does not define too much standards like SOAP.
- SOAP requires more bandwidth and resource than REST. REST requires less bandwidth and resource than SOAP.
- SOAP defines its own security. RESTful web services inherits security measures from the underlying transport protocol.
- SOAP permits XML data format only. REST permits different data format such as Plain text, HTML, XML, JSON, etc.
- SOAP is less preferred than REST. REST more preferred than SOAP.
- SOAP cannot be cached while REST can be cached.

→ **1.5 marks x 4**

(d) Briefly discuss the functionality of following **HTTP Verbs**.

(4 marks)

DELETE:  
PATCH:  
PUT:  
HEAD:

- PUT is used to update a certain record (data) in the server side. → **1 marks**
- DELETE is used to delete a certain record (data) in the server. → **1 marks**
- PATCH is used to update a certain field of a record (data) in the server side. → **1 marks**
- HEAD is used to obtain the header information of a particular web resource. → **1 marks**

- (e) The following code segment was taken from a Restful web application.

(5 marks)

```
@RequestMapping(value="/comment/findCommentsByuserId/{token}", method=RequestMethod.GET)
public @ResponseBody List<CommentHttp> findCommentsByuserId(@PathVariable String token,
HttpServletResponse response) {
...
...
}
```

Describe the meaning or the intended action of the following terms? (5 marks)

- i. @RequestMapping :
- ii. RequestMethod.GET :
- iii. @ResponseBody:
- iv. @PathVariable:
- v. HttpServletResponse :

#### **ANSWER IN THIS BOX**

@RequestMapping: This annotation will map a particular Java method to a particular HTTP request URL. Once the URL is called, the method will be triggered automatically.→ **1 marks**

RequestMethod.GET: This defines the Type of the HTTP method which will be used for the communication between client and the server. In Restful applications GET methods are used to receive a particular resource.→ **1 marks**

@ResponseBody: This defines the return data structure for the response.→ **1 marks**

@PathVariable: URL request variables that deliver the required information for the server method from the client.→ **1 marks**

HttpServletResponse response: This is the initialization of the response message which will be send to the client from the server.→ **1 marks**

4)

- (a) List **four** differences between **CORBA** and **RMI** in developing client server applications.

(6 marks)

#### **ANSWER IN THIS BOX**

- RMI is a Java-specific technology and runs on top of JVM. CORBA has implementations for many languages as well as CORBA services can be executed on many different platforms.
- RMI uses Java interfaces. CORBA uses IDL (Interface Definition Language) to separate interface from implementation. Therefore, CORBA developers can create different implementations based on the same interface.

- CORBA objects are not garbage collected. Once a CORBA object is created, it continues to exist until the programmer decides to get rid of it. RMI objects are garbage collected automatically.
- In CORBA, the data types do not always map exactly to the types used by the programming language (e.g. a long in IDL is an int in Java).
- RMI programs can download new classes from remote JVMs. CORBA doesn't have a code sharing mechanism.
- RMI can be operated over IIOP (the protocol used by CORBA). Through RMI-IIOP it is possible to achieve interoperability.
- CORBA supports in and out parameters, while RMI does not since local objects are passed by copy and remote objects are passed by reference.

→ **1.5 marks (for any correct answer) x 4**

(b) What are the roles played by Java **RMI Registry in Java RMI programming?**

**(3 marks)**

**ANSWER IN THIS BOX**

RMI Registry acts a broker (service directory) between RMI servers and the clients. → **1 marks**

RMI servers register their services in the RMI registry. It is not necessary for a RMI client to know the location of individual servers, and does a lookup on the RMI Registry for the service it needs. The registry, being a naming directory returns the appropriate handle to the client to invoke methods on the server. → **2 marks**

(c) Explain the **functionality** of each of the following **code segments / commands**. (4 marks)

- I. `import java.rmi.registry.Registry;`
- II. `public interface Hello extends Remote {}`
- III. `Hello stub = (Hello) registry.lookup("Hello");`
- IV. `start rmiregistry`

**ANSWER IN THIS BOX**

- I. `import java.rmi.registry.Registry;`  
'public interface Registry' is a remote interface to a simple remote object registry that provides methods for storing and retrieving remote object references bound with arbitrary string names. The bind, unbind, and rebind methods are used to alter the name

	<p>bindings in the registry, and the lookup and list methods are used to query the current name bindings.→ <b>1 marks</b></p> <p>II. public interface Hello extends Remote {  'Remote' interface declares the methods that will be called remotely. All methods described in the Remote interface must list RemoteException in their throws clause.  → <b>1 marks</b></p> <p>III. Hello stub = (Hello) registry.lookup("Hello");  Get a reference to the remote object implementation from the server host's rmiregistry.  → <b>1 marks</b></p> <p>IV. start rmiregistry  This command will start the RMI registry inside the JVM.→ <b>1 marks</b></p>	
(d)	<p>Explain <b>dynamic invocation</b> procedure in CORBA.</p> <p style="text-align: right;"><b>(4 marks)</b></p>	
	<p><b><u>ANSWER IN THIS BOX</u></b></p> <p>When using dynamic invocation, the CORBA client application can dynamically build operation requests for a CORBA object interface that has been stored in the Interface Repository.  → <b>2 marks</b></p> <p>This supports deferred synchronous communication, where the CORBA client application sends the request and it is free to do other work. Dynamic Invocation also enables a CORBA client application to invoke a method on a CORBA object whose type was unknown at the time the CORBA client application was written.→ <b>2 marks</b></p>	
(e)	<p>Describe the functionalities provided by <b>POA</b> in CORBA.</p> <p style="text-align: right;"><b>(4 marks)</b></p>	
	<p><b><u>ANSWER IN THIS BOX</u></b></p> <p>In CORBA, object adapter connects a request using an object reference with the proper code to service that request. → <b>2 marks</b></p> <p>The Portable Object Adapter ( POA) is a particular type of object adapter that provides following functionalities:</p> <ul style="list-style-type: none"> <li>▪ POA allows programmers to construct object implementations that are portable between different ORB products.</li> <li>▪ It supports for objects with persistent identities.</li> <li>▪ It supports transparent activation of objects.</li> <li>▪ Associate policy information with objects.</li> <li>▪ Allow multiple distinct instances of the POA to exist in one ORB.</li> </ul> <p>→ <b>1 marks x 2</b></p>	

(f) State the differences between COM and DCOM.

(4 marks)

**ANSWER IN THIS BOX**

COM (Component Object Model) is a computing environment where reusable functions or the business logic of an application are bundled as a component (as a Dynamic Link library or another local process) to be invoked whenever necessary by the presentation layer of the application. COM is executed at a local level, at the client's machine. On the other hand, DCOM (Distributed Component Object Model) runs at the server end, where the client passes instructions to the DCOM object and get it executed at the server over the network.

→ **4 marks**

\*\*\*\*\*