



The University of Fiji

School of Science and Technology



Department of Computer Science/ Information Technology

ITC333

Distributed Systems Technology

Semester I, 2014

FINAL EXAMINATION QUESTION PAPER

Time Allowed: 3 Hours [10 Minutes Extra Allowed For Reading the Paper]
Weighting: 50%
Total Marks: 100 marks.

45 Short Answer Questions

Introduction	-	14 marks
Architectures		10
Processes		9
Communication		9
Naming		9
Synchronization		9
Consistency and Replication		11
Fault Tolerance		13
Security		12
Programming		4
Total		100 marks

Instructions

- Write all your answers in the Answer Book provided.
- Show your answers clearly in each question to gain maximum marks.
- You may ask for extra sheets from the supervisors if the need arises.
- Do not forget to write your name and ID number on all the extra sheet(s) attached.

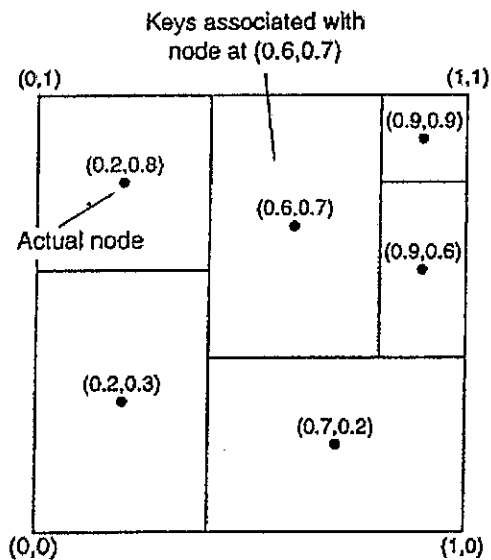
Introduction [14 marks]

1. There are many forms of transparency in a distributed system. State **four** forms of transparency. Further, state descriptions for any two out of the four.
[3 marks]
2. Why is it sometimes so hard to hide the occurrence and recovery from failures in a distributed system?
[1.5 marks]
3. Why is it not always a good idea to aim at implementing the highest degree of transparency possible?
[1 mark]
4. What is an open distributed system and what benefits does openness provide?
[2 marks]
5. Describe precisely what is meant by a scalable system.
[2 marks]
6. Scalability can be achieved by applying different techniques. What are these techniques? (State at least two)
[1 mark]
7. Draw a labeled diagram showing a layered architecture for grid computing systems.
[2 marks]
8. What are three requirements for pervasive applications in distributed pervasive systems?
[1.5 marks]

Architectures [10 marks]

9. Draw a simple diagram of a component-based architecture?
[2 marks]
10. What is three-tiered client-server architecture?
[1.5 marks]

11. Consider the Content Addressable Network (CAN) given below.



How would you route a message from the node with coordinates (0.2, 0.3) to the one with coordinates (0.9, 0.6)?

[2 marks]

12. Considering that a node in CAN knows the coordinates of its immediate neighbors, a reasonable routing policy would be to forward a message to the closest node toward the destination. How good is this policy? Explain.

[1.5 marks]

13. Not every node in a peer-to-peer network should become superpeer. State three reasonable requirements that a superpeer should meet?

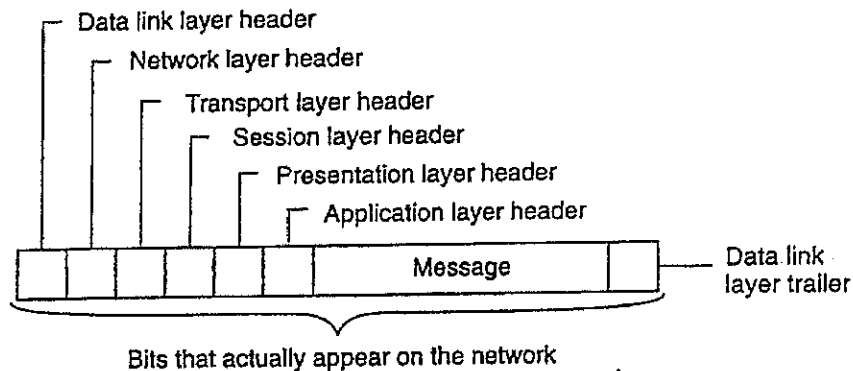
[3 marks]

Processes [9 marks]

14. State two reasons why it makes sense to limit the number of threads in a server process?
[2 marks]
15. Multithreaded file server is better than a single-threaded server and a finite-state machine server. Are there any circumstances in which a single-threaded server might be better?
Give an example.
[1 mark]
16. What is the major drawback of a user-level thread?
[1 mark]
17. Describe a simple scheme in which there are as many lightweight processes as there are runnable threads.
[2 marks]
18. Explain the principles of "TCP handoff".
[3 marks]

Communication [9 marks]

19. In many layered protocols, each layer has its own header, as shown below.



Surely it would be more efficient to have a single header at the front of each message with all the control in it than all these separate headers. Why is this not done?

[2 marks]

20. Consider a procedure *incr* with two integer parameters. The procedure adds one to each parameter. Now suppose that it is called with the same variable twice, for example, as *incr(i, i)*. If *i* is initially 0, what value will it have afterward if call-by-reference is used? How about if copy/restore is used?

[2 marks]

21. Summarize remote procedure call (RPC) in ten steps.

[5 marks]

Naming [9 marks]

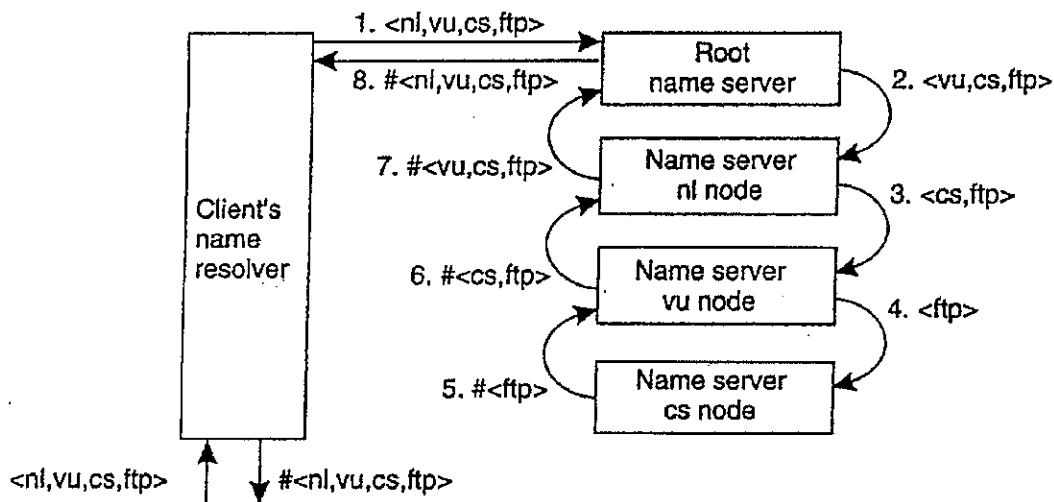
22. A directory node contains a directory table in which an outgoing edge is represented as a pair. What two pieces of information are in this pair?

[1 mark]

23. Give an example of where an address of an entity *E* needs to be further resolved into another address to actually access *E*.

[2 marks]

24. What is a major drawback of recursive lookups when resolving a key in a DHT-based system?



[3 marks]

25. What information is required to mount a *foreign name space* in a distributed system?

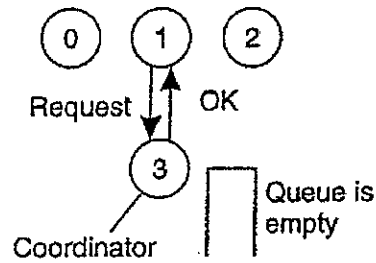
[3 marks]

Synchronization [9 marks]

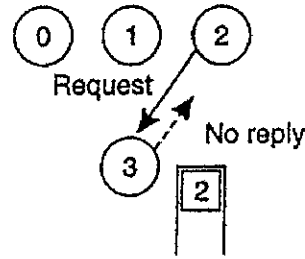
26. To achieve totally-ordered multicasting with Lamport's timestamps, is it strictly necessary that each message is acknowledged? Explain your answer.

[3 marks]

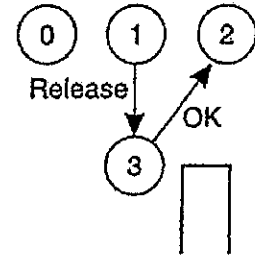
27. Discuss centralized algorithm as in the figures a, b, and c below.



(a)



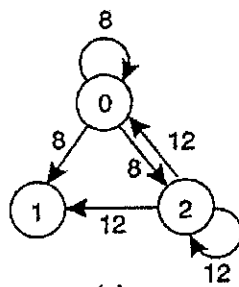
(b)



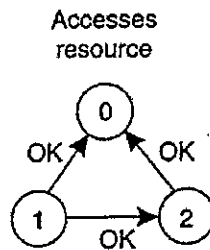
(c)

[3 marks]

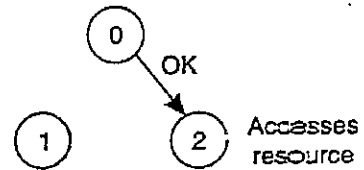
28. Describe how the distributed algorithm given below works.



(a)



(b)



(c)

[3 marks]

Consistency and Replication [11 marks]

29. What are the two primary reasons for replicating data?

[2 marks]

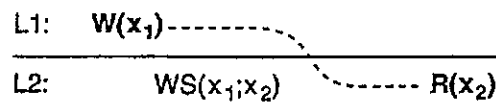
30. Explain what the main reason is for actually considering weak consistency models.

[1 mark]

31. In consistency models, why is there a need for a contract between the software and data store.

[2 marks]

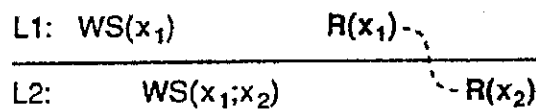
32. A data store that provides read-your-writes consistency is shown below.



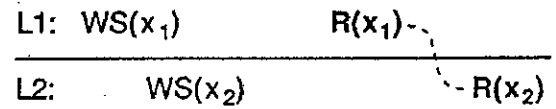
What condition does it uphold?

[2 marks]

33. Explain figures (a) and (b) below in relation to monotonic reads.



(a)

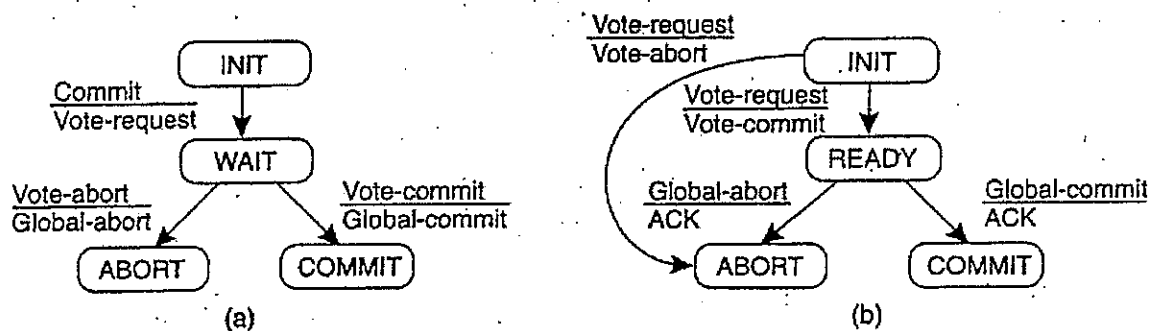


(b)

[4 marks]

Fault Tolerance [13 marks]

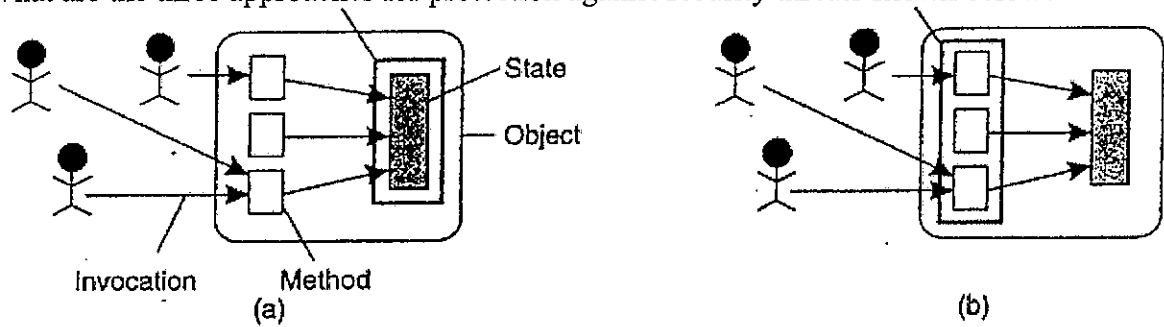
34. What makes the fail-stop model in the case of *crash failures* so difficult to implement? [1 marks]
35. The key technique for masking faults is to use redundancy. There are three kinds of redundancy – time, information, and physical. Explain physical redundancy using an example [2 marks]
36. Draw a labeled simple Triple Modular Redundancy (TMR) circuit diagram for providing fault tolerance. [2 marks]
37. Receiver-based message logging is generally considered better than sender-based logging. Why? [1 mark]
38. Explain the two-phase commit steps for the coordinator and the participant.



39. In the two-phase commit protocol, why can blocking never be completely eliminated, even when the participants elect a new coordinator? [1 marks]
40. In our explanation of three-phase commit, it appears that committing a transaction is based on majority voting. Is this true? Explain your answer. [2 marks]

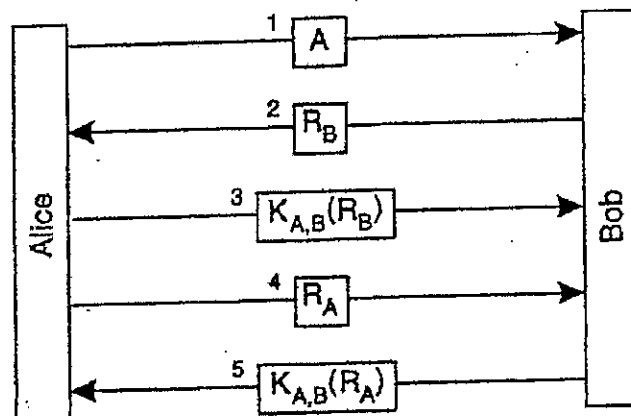
Security [12 marks]

41. What are the three approaches for protection against security threats shown below?



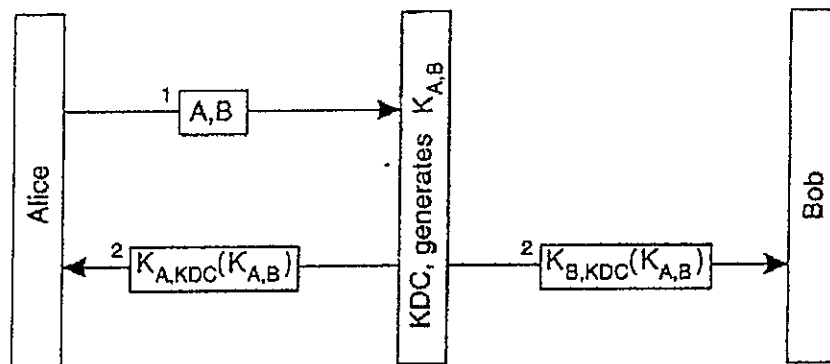
[1 mark]

42. Explain the figure below on authentication based on shared secret key.



[5 marks]

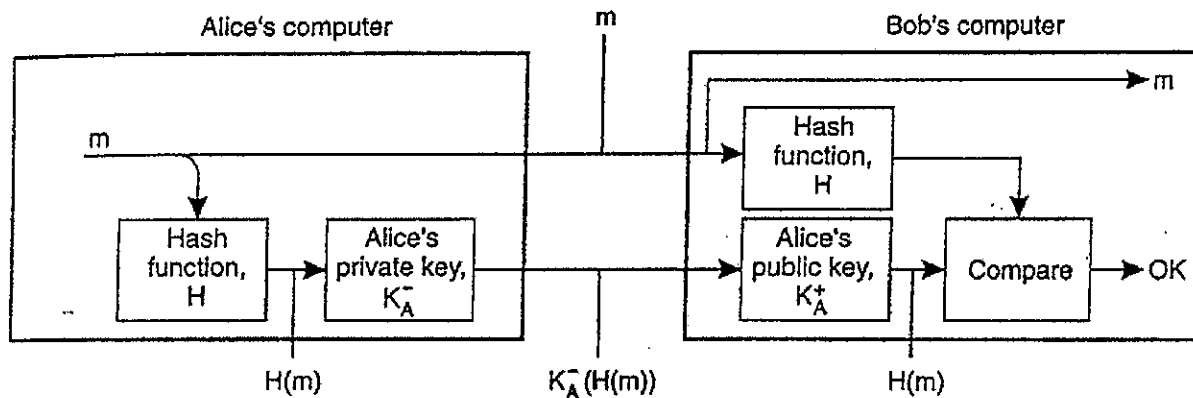
43. If Alice wants to set up a secure channel with Bob, she can do so with the help of a (trusted) KDC.



Why is it not necessary in the figure above for the KDC to know for sure it was talking to Alice when it receives a request for a secret key that Alice can share with Bob?

[2 mark]

44. Explain digitally signing a message using a message digest.



[4 marks]

Programming [4 marks]

45. Write the missing codes to complete the program to simulate the functioning of Lamport's logical clock in C++. A sample output is given below.

```
Enter the no. of physical clocks: 2
No. of nodes for physical clock 1 :2
Enter the name of process: a
Enter the name of process: b

No. of nodes for physical clock 2 :2
Enter the name of process: c
Enter the name of process: d

Press a key for watching timestamp of physical clocks

Physical Clock 1
Process a has P.T. :1
Process b has P.T. :9

Physical Clock 2
Process c has P.T. :4
Process d has P.T. :5

Press a key for watching timestamp of logical clocks
Logical Clock Timestamp for process a :9
Logical Clock Timestamp for process c :13
Logical Clock Timestamp for process d :21
Logical Clock Timestamp for process b :29
```

[4 marks]

```
#include <conio.h>
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int i,j,k;
    int x=0;
    char a[10][10];
    int n,num[10],b[10][10];

    printf("Enter the no. of physical clocks: ");
    scanf("%d",&n);

    for(i=0;i<n;i++)
    {
        printf("\nNo. of nodes for physical clock %d :",i+1);
        scanf("%d",&num[i]);
        x=0;
```

```

        for(j=0;j<num[i];j++)
        {
            printf("Enter the name of process: ");
            scanf("%s",&a[i][j]);
            b[i][j]=x + rand() % 10;
            x=b[i][j]+1;
        }
    }

    printf("\nPress a key for watching timestamp of physical clocks ");
    getch();

    for(i=0;i<n;i++)
    {
        printf("\n\nPhysical Clock %d",i+1);
        for(j=0;j<num[i];j++)
        {
            printf("\nProcess %c",a[i][j]);
            printf(" has P.T. :%d ",b[i][j]);
        }
    }

    printf("\n\nPress a key for watching timestamp of logical clocks \n");
    getch();

    x=0;
    for(i=0;i<10;i++)
        for(j=0;j<n;j++)
            for(k=0;k<num[j];k++)
                if (b[j][k]==i)
                {


|  |
|--|
|  |
|  |
|  |
|  |


                }
    }
    getch();
}

```

THE END