1. The ostrich algorithm provides an excellent mechanism to prevent deadlocks. explains , true or false.

**Answer:**

Excellence of ostrich algorithm depends on deadlock situation.

The ostrich algorithm says, stick your head in the sand and pretend there is no problem at all. Mathematicians find it totally unacceptable and say that deadlocks must be prevented at all costs. Engineers ask how often the problem is expected, how often the system crashes for other reasons, and how serious a deadlock is.

If deadlocks occur on the average once every five years and system fail occur once a week, most engineers would not be willing to pay a large penalty in performance or convenience to eliminate deadlocks. Hence its False .

1. Executable file is a ………. File. explain.

Answer:

Executable file is a Binary File. It has five sections: header, text, data, relocation bits, and symbol table.

* The header starts with a so-called magic number, identifying the file as an executable file (to prevent the accidental execution of a file not in this format).
* Then come the sizes of the various pieces of the file, the address at which execution starts, and some flag bits.
* Following the header are the text and data of the program itself. These are loaded into memory and relocated using the relocation bits.
* The symbol table is used for debugging.

1. Parallel and distributed computer systems can be divided into \_\_\_\_\_\_\_\_\_\_\_\_\_\_ multiprocessors and \_\_\_\_\_\_\_\_\_\_\_\_\_ multicomputers in terms of memory.

Answer:

Parallel and distributed computer systems can be divided into \_\_\_\_\_\_4\_\_\_\_\_\_\_\_ multiprocessors and \_\_\_\_\_\_7\_\_\_\_\_\_\_ multicomputers in terms of memory.

Multiprocessors:

* Multiprocessor Hardware
* Multiprocessor Operating system types
* Multiprocessor synchronization
* Multiprocessor scheduling

Multicomputers:

* Multicomputer Hardware
* Low-Level communication software
* User-Level Communication software
* Remote Procedure Call
* Distributed shared Memory
* Multicomputer scheduling
* Load balancing

1. What is a “Name server” and how is it used in client-server model?

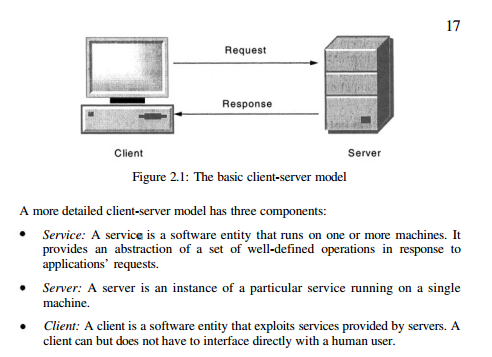
Answer:

Name Server:

A nameserver is a computer that is permanently connected to the Internet and translates domain names into IP addresses (or vice versa), enabling you to enter www.example.com instead of 194.63.248.47

The **client–server model** is a [distributed application](https://en.wikipedia.org/wiki/Distributed_application) structure that partitions tasks or workloads between the providers of a resource or service, called [servers](https://en.wikipedia.org/wiki/Server_(computing)), and service requesters, called [clients](https://en.wikipedia.org/wiki/Client_(computing)).

The Domain Name System is maintained by a [distributed database](https://en.wikipedia.org/wiki/Distributed_database) system, which uses the [client–server model](https://en.wikipedia.org/wiki/Client%E2%80%93server_model). The nodes of this database are the [name servers](https://en.wikipedia.org/wiki/Name_server). Each domain has at least one authoritative DNS server that publishes information about that domain and the name servers of any domains subordinate to it.



1. What is redundancy and why is it used in a distributed system ? explain.

Answer.

**Redundancy** is the duplication of critical [components](https://en.wiktionary.org/wiki/Component) or functions of a system with the intention of increasing reliability of the [system](https://en.wikipedia.org/wiki/System), usually in the form of a backup or [fail-safe](https://en.wikipedia.org/wiki/Fail-safe).

In Distributed system, the information routers exchange information frequently. To make is reliable and guaranteed delivery, even in the face of crashes, It is necessary to store old tuples in case they are needed later.

1. Match each of the following deadlock condition with its correct prevention approach. Explain how your choice prevents deadlock condition.

|  |  |
| --- | --- |
| I. Mutual Exclusion - 3 | 1. Request all resources initially |
| II. Hold and wait - 1 | 2. Order resources numerically |
| III. No Preemption - 4 | 3. Spool everything |
| Iv. Circular wait - 2 | 4. Take resources away |

**Answer:**

Mutual Exclusion -> Spool everything

Hold and wait -> Request all resources initially

No Preemption -> Take resources away

Circular wait -> Order resources numerically

1. A unix file system has 1 KB blocks and 4-byte disk address. What is the maximum file size if i-nodes contain 10 direct entries and one single, double and triple indirect entry each ?
   1. 16GB
   2. 4GB
   3. 40GB
   4. 1.6GB

Answer: 16GB

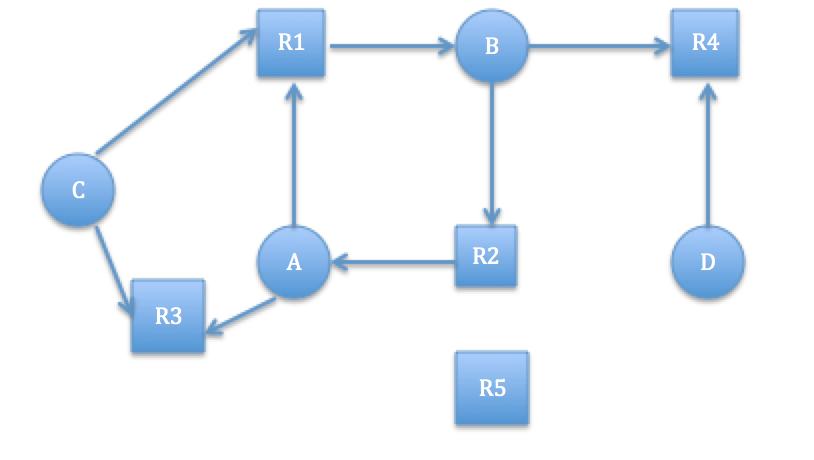
The i-node holds 10 pointers. The single indirect block holds 256 pointers. The double indirect block is good for 2562 pointers. The triple indirect block is good for 2563 pointers. Adding these up, we get a maximum file size of 16,843,018 blocks, which is about 16.06 GB.

1. When a procedure is transported from one machine and placed on another to be called by RPC, some problems can occur. In the text, we pointed out four of these: pointers, unknown array sizes, unknown parameter types, and global variables.
   1. Explain these in short.
   2. An issue not discussed is what happens if the (remote) procedure executes a system call. What problem might that cause and what might be done to handle it ?

Answer:

1. A system has four processes A,B, C and D and five resources R1, R2, R3, R4, R5. the states of the resources owned and requested are shown below. Draw the resource graph and show if system is deadlocked. Also suggest a way to recover from it.
   1. Process A owns R2 and requests R1 and R3
   2. Process B owns R1 and request R2 and R4
   3. Process C owns nothing but requests R1 and R3
   4. Process D requests R4

Answer:



The graph contains one cycle that is A - R1 - B - R2 - A and process A,B are in deadlocked.

To Recover from deadlock there are following option we can choose -

Recovery through Preemption:

In some cases it may be possible to temporarily take a resource away from its current owner and give it to another process. In many cases, manual intervention may be required, especially in batch processing operating systems running on mainframes.

The ability to take a resource away from a process, have another process use it, and then give it back without the process noticing it is highly dependent on the nature of the resource. Recovering this way is frequently difficult or impossible. Choosing the process to suspend depends largely on which ones have resources that can easily be taken back.

Recovery through Rollback:

If the system designers and machine operators know that deadlocks are likely, they can arrange to have processes “**check pointed”** periodically. When a deadlock is detected, it is easy to see which resources are needed. To do the recovery, a process that owns a needed resource is rolled back to a point in time before it acquired that resource by starting one of its earlier checkpoints.

Recovery through Killing Processes:

The crudest, but simplest way to break a deadlock is to kill one or more processes. One possibility is to kill a process in the cycle. With a little luck, the other processes will be able to continue. If this does not help, it can be repeated until the cycle is broken. It might happen, process not in the cycle can be chosen as the victim in order to release its resources. Sometime, two deadlock can be happen and killing one process not able to recover deadlocks.

1. suppose that we have free segments with sizes: 6, 17, 25, 14 and 19. Place a program with size 13kb in the free segment using
   1. First-fit
   2. Best-fit
   3. Worst fit

Answer:

First-fit:

The simplest algorithm is first fit. The memory manager scans along the list of segments until it finds a hole that is big enough. The hole is then broken up into two pieces, one for the process and one for the unused memory, except in the statistically unlikely case of an exact fit. First fit is a fast algorithm because it searches as little as possible.

First fit takes 17 KB

Best-fit:

A well-known and widely used algorithm is best fit. Best fit searches the entire list, from beginning to end, and takes the smallest hole that is adequate. Rather than breaking up a big hole that might be needed later, best fit tries to find a hole that is close to the actual size needed, to best match the request and the available holes. Best fit is slower than first fit because it must search the entire list every time it is called. Somewhat surprisingly, it also results in more wasted memory than first fit.

Best fit takes 14KB

Worst fit:

worst fit, that is, always take the largest available hole, so that the new hole will be big enough to be usefuL Simulation has shown that worst fit is not a very good idea either. And its slow as well because it need to find large free segments among entire list of segments.

Worse fit takes 25KB

1. Select your favorite topic covered in this course and relate with its corresponding SCI principle.

Answer: