

2022

Q. What are different types of kernels?

⇒ The different types of kernels are:

- o Micro-kernel
- o monolithic-kernel
- o Nano-kernel
- o Embedded-kernel
- o Hybrid-kernel

Q. What kinds of operations can be performed on Semaphores

- o wait() - Decrements the value of Semaphore.
- o signal() - Increments the value of Semaphore.

Q. What exactly do you mean when you say "zombie process"?

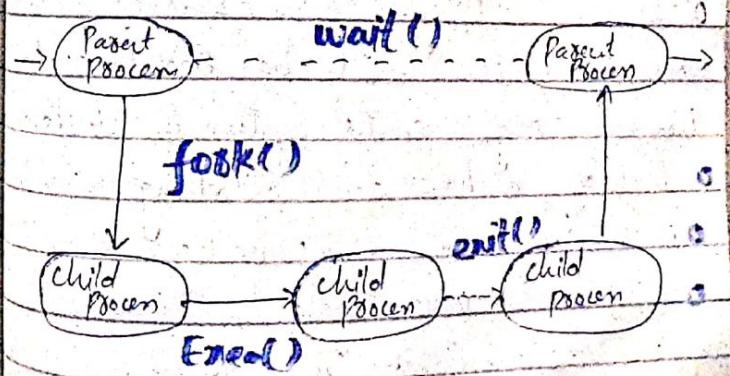
o A zombie process, also called a "defunct process", is a process that has finished its execution but still has an entry in the process table.

o This happens when the parent process fails to collect the exit status of

o Terminated child.
The zombie process usually occurs in the child process. very short time a process is zombie.

o After the process has completed all its tasks it reports the parent process that it has to about terminate.
Zombie is unable to terminate itself because it's treated as a dead process. So, parent process needs to execute to terminate the command to terminate the child.

o Zombie process has controlling terminals. Too many zombie process can be harmful. So handling of the wait system call is necessary to avoid ~~zombie process~~.



Dolphin process:-

- o A child process that remains running even after its parent process is terminated or completed without waiting for the child process execution is called an orphan.
- o A process becomes an orphan unintentionally.
- o Sometimes intentionally becomes orphans due to long - running time to complete the assigned task without user attention.
- o The Dolphin process has controlling terminals.
- o To communicate the dolphin process use the SIGPUP signal.

Daemon process:-

- o Daemon process are started working when the system will be booted up and terminate only when the system is shutdown.
- o it does not have a controlling terminal, it always runs in the background.
- o Daemon process state is indicated by ? in the field of its column.

Q. Why SJF cannot handle real time environment be you don't have enough memory when

PSJF algorithms etc especially in non-real time environments where the

SJF not suitable for real time environments.

not because of lack of SJF handling of pre-emptability.

o handling of variability and potential for starvation.

Realtime systems require scheduling algorithms specially designed to meet their requirements and prioritize tasks on their deadlines or priorities.

Q. Suppose there is an organization which hired a person dividing tasks among others, which multiprocessor environment does this organization depict?

- In master-slave multiprocessing environment one processor is work as master and assign tasks to other processors.

So, ~~master slave~~ multiprocessor environment is best for this organization.

Q. What is processor affinity?

- Processor affinity is also called CPU pinning.
- It enables applications to bind or unbind a process or thread at a specific core or to a range of cores or CPUs.
- The OS ensures that a given thread executes only on the assigned cores, or each time it is scheduled, if it was pinned to a core.
- It also called cache affinity and ~~is implemented in hardware~~ by a processor specific L1, L2, L3 cache.
- It is modification of symmetric multiprocessing with OS initiated scheduling, such as scheduling by gainsharing.

What is hard affinity & soft affinity?

These are two main types of process affinity.

- 1) Soft affinity
- 2) Hard affinity

Soft affinity:

It refers to the OS policy of keeping a process running on the same processor, and it will not issue that it will do so.

Hard affinity:

In hard affinity a process specifies that it is not to be moved between processes.

It enables a process to define a subset of processors on which it can execute.

Some OS like Linux implement affinity and offer system calls such as set scheduling policies.

What is "reference bit" and what is "dirty bit"?

Reference bit

- o Reference bit specifies whether that page has been referenced in the last clock cycle or not.

Dirty bit

- o Also known as modified bit. Specifies whether that page has been modified since it was last written.
- o It is also known as dirty bit.
- o If the page has been referenced then it is modified then it is set, this bit is set to 1 otherwise.
- o It helps to avoid unnecessary writes.

R. What is the concept of dynamic linking?

- o Dynamic linking is the process of writing only the name of the shared library module in program.

- o This dynamically linked file includes only the name of calling program.

R- Define file system?

Name file system

- o File System is an integrated proto OS.
- o A file System is a data structures that stores data and information on storage devices (hard drives, floppy disc etc) making them easily retrievable.

- o Different file systems, but all have different features.

Pros

- o Organization
- o Data protection
- o Improved performance

Cons

- o Incompatibility issues
- o Disk space overhead
- o Vulnerability

Q. What is disk controller?

Ans

The disk controller is the circuit that regulates the operation of floppy disk and other disk drives.

It is like a plug-in circuit board.

It controls the flow of data between computer's storage device and its memory.

Act as interface b/w the storage unit and the OS, controlling the read and write operations and providing reliable data transfer.

- o Protection
- o Clean

Functions of memory manager

- o Command translation
- o Data transfer
- o Error correction
- o Interrupt handling
- o Power management
- o Disk geometry translations in local.

Q. Define garbage collection?

- o Garbage collection is a dynamic technique for memory management and heap allocation that examines and identifies dead memory blocks before deallocated from reuse.
- o Primary purpose of garbage collections is to reduce memory blocks.
- o Garbage collection schemes
 - o Reference Counting
 - o Mark and Sweep
 - o Stop and Copy
- o Running program is affected by collection.

• It lists the security effects of the security attack.

- The computer runs slower than in.
- The application software very slow.
- Devices cannot be accessed and clock is stopped.

Stages of performed during

Content switching:

Sequence of actions

- 1) OS takes control (through interrupt)
- 2) Saves Content of running process in the Process PCB.
- 3) Reload Content of new process from the new process PCB.
- 4) Return control to new Process.

Q. Discuss different threading issues? (PP-2022)

Definition:

Thread is an execution unit that consists of its own program counter, a stack, and a set of registers, where the program counter mainly keeps track of which instruction to execute next, a set of registers mainly holds current working variables, and a stack mainly contains the history of execution.

Threads are a popular way of improving the performance of an application through parallelism.

Issues:-

Threading issues mostly occur in multi-threading environment.

fork() & exec() system calls:

The fork() and exec() are the calls.

o The fork() call creates a duplicate process of the parent that invokes fork().

- o The new duplicate process is called child process and the process that invoke the fork() is called the parent process.

- o Both the parent process and the child process continue their execution from the instruction that is just after the fork().

Issue:

- o If a thread of the multi-threaded program has invoked the fork(). So the fork() would create a new duplicate process.
- o Whether the new duplicate process created by fork() will duplicate all the threads of the parent process or duplicate process would be a single threaded process.

exec() calls:

- o exec() system call when invoked replaces ~~the program along with all its threads~~ with the program that is specified in the parameters to exec(). Typically the exec() system call is lined up after the fork() system call.

Issue:

If the exec() system call is after the fork() call is duplicating all the system call in the child process of parent process. Then the entire process by fork() is replaced. The exec() system call exec() in the parameters with the process provided.

Thread termination:

- o Thread that are no longer required can be cancelled by another thread in the middle of its execution.
- o Asynchronous cancellation
- o Deferred cancellation

Asynchronous cancellation:-

- o It terminates the target thread immediately.
- o Allocation of resources and inter thread transfers may be challenging for the system.

Deferred Cancellation:

⇒ Deferred Cancellation allows the target thread to periodically check if it should be canceled.

⇒ A flag is set that indicating the thread should cancel itself when it is feasible.

(3) Signal handling:

- A signal is used to notify a process that a particular event has occurred.

- A signal may be received either synchronously or asynchronously depending on the source and the reason for the event.

- A signal may be handled by one of two possible handlers:

- A default signal handler

- A user-defined signal handler

- Every signal has a default signal handler that the kernel runs when handling that signal.

- A user-defined signal handler that is called to handle the signal.

Some signals are handled by terminating the program and some are simply ignored whenever a multithreaded process receives a signal then to what thread should that signal be conveyed. These are following four main options:

(1) Signal delivers to the thread to which the signal applies.

(2) Signal delivers to each and every thread in the process.

(3) Signal delivers to some of the threads in the process.

(4) Assign a particular thread to receive all the signals in a process.

(4) Security issues:

- These can be security issues because of extensive sharing of resources between multiple threads.

(5) Thread local storage:

The benefit of using threads in the first place is that most data is shared among the threads but sometimes threads

Practice Questions

Consider the following snapshot of a system:

	Allocation				Main				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
T ₀	0	0	1	2	0	0	1	2	1	5	2	0
T ₁	1	0	0	0	1	7	5	0	1	5	3	2
T ₂	1	3	5	4	2	3	5	6	2	8	8	6
T ₃	0	6	3	2	0	6	5	2	3	8	8	6
T ₄	0	0	1	4	0	6	5	6	3	8	8	6
									3	14	12	10

Apply banked algorithm to find need matrix?

Is the system in safe state?

If a request from thread T₁ arrives for (0,4,2,0), can the demand be granted immediately?

Need matrix

	A	B	C	D
T ₀	0	0	0	0
T ₁	0	7	5	0
T ₂	1	0	0	2
T ₃	0	0	2	0
T ₄	0	6	4	0

- Major libraries of threads are pThreads, Win32 and java which provide support for thread specific which is called TLS thread local storage.

(b) Scheduled Activation:

Communication

- A final issue between the kernel and the thread library.
- Numerous implementation of threads provide a virtual processor as an interface between user and the kernel thread specifically for two tier model.
- The virtual processor is called low weight process.

kernel thread ~~multi processor~~ to one correspondence.

The available numbers of kernel threads can be changed dynamically. The OS is used to schedule on the basis of system.

3 ← user thread

LWP ← lightweight process

X "kernel thread"

Banker's algorithm

(b) Safe sequence

Indicates availability of all resources

$T_0, T_1, T_2, T_3 \rightarrow$ - message to

old bank . . . loan . . .

o Yes, the System is in Safe state.

(c)

Yes, the request can be granted immediately.
The value of available is then $(1, 1, 0, 0)$

Object having a running job assigned to it

can be

safe state in it satisfies all its

IT bank must know if it

exists not enough memory

available

simply bank

Q.What is ISR?

ISR stands for interrupt service routine.

An ISR also called an interrupt handler.

It is a software routine that handles

invokes in response to an interrupt.

ISR examines an interrupt and determines

how to handle it executes the handling

and then returns a logical interrupt value.

If no problem handling is sematical the ISR notifies the kernel with a return value

example:

An ISR just handles keyboard

events such as pressing or releasing

key each time a key is pressed

The ISR processes the input

Q.What is kernel structure?

A kernel structure is the control

module of an OS. It is the one

structure that loads first and remains

in the memory.

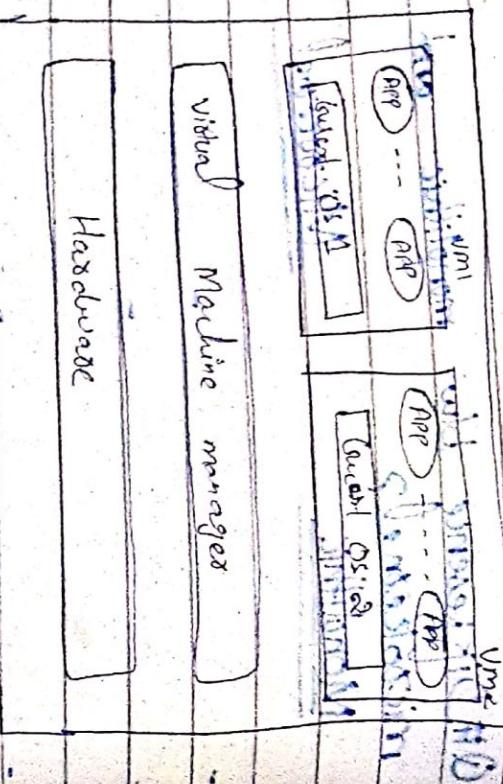
If this was system program & hosted

This kernel uses system calls

to perform all its functions like

What is virtual machine?

- A virtual machine is a virtual environment which functions as a virtual computer system with its own processing unit, memory, system interface and storage, created at hardware interface and storage systems.
- VMs are isolated from the rest of system, and multiple VMs can exist on a single piece of hardware like server.



Global page replacement strategy

Global page replacement

If a process needs a page which is not in the memory, it can undergo a page fault which is allocated to another process. So, OS will find a page which is available scanner.

Process selects a replacement frame from the set of all frames. One process can take frame from another.



Local page replacement

If a process wants to bring its own page, then each process selects only its own set of allocated frames.

Q: Difference between long term

1. Short term and medium term schedules?

Long-Term schedules	Short-term Scheduling	Medium-Term Scheduling
1) It is a Job scheduler.	It is a CPU Scheduler.	It is a process swapping scheduler.
2) Speed is lesser than Short term scheduler.	Speed is fastest among all the two.	Speed is in between both short and long term schedules.
3) It controls the degree of programming.	It provides less control over the degree of multiprogramming.	It reduces the degree of multiprogramming.
4) It is almost absent or minimal in time sharing system.	It is also minimal in time sharing system.	It is a part of time sharing systems.
5) It selects processes from pool & loads them into memory for execution.	A selected those processes which are ready to execute.	It can reintroduce the process into memory and execution can be

Q: What is busy waiting?

Busy waiting is a technique in which a process repeatedly checks to see if a condition is true instead of blocking & spin lock use this technique for the purpose of checking if a lock is available.

Q: What is difference between bounded waiting and unbounded waiting with example?

⇒ "Pogoers" means that if a process is not using the critical section, then it should not stop any other process from entering it. In other words, any process can enter a critical section if it is free.

Bounded waiting means that each process must have a limited waiting time. It should not wait endlessly to enter the critical section. No process should wait for a resource for infinite amount of time.

Difference b/w Caching & buffering

<u>Buffering</u>	<u>Caching</u>
<u>Used for:</u>	
o it matches the speed of data stream b/w the senders and receivers.	o it improves the access speed of the frequently used data.
<u>Policy:</u>	
o it uses FIFO.	o it follows LRU.
<u>Stores what?</u>	
o Original copy of data.	Original copy of data.
<u>Usage:</u>	
o used for i/p and o/p buffers.	o used for reading & writing process from disk.
<u>Location:</u>	
o implemented in RAM.	o In RAM as well as disk.