fork() is a system call in Unix-based operating systems that creates a new process by duplicating the calling process.

The heap is a region of a program's memory that is used for dynamic memory allocation. It is a region of memory that is managed by the operating system's memory allocator.

Device-status table contains entry for each I/O device indicating its type, address, and state

iOS asks apps to voluntarily relinquish allocated memory:

Read-only data thrown out and reloaded from flash if needed

 Failure to free can result in termination

Parallelism implies a system can perform more than one task simultaneously

is many-to-many used in thread? False ( it might be true )

compare and swap starts with .. ? 0

If we make the lock = 0, what will happen? only one process will enter, the rest will be dead locked.

Semaphore starts with int = 0.

Semaphore uses acquire and release. False.

Starvation means that jobs are waiting in waiting queue infinitely. True.

Deadlock – two or more processes are waiting indefinitely for an event that can be caused by only one of the waiting processes.

atomic operation is an operation that appears to occur instantaneously and indivisibly to the rest of the system.

The thread library used in java called Bthread. <mark>False</mark>.

<mark>Lightweight Process (LWP)</mark> : Typically use an intermediate data structure between user and kernel threads.

<mark>A PCB</mark> is a data structure that is used by an operating system to store information about a process or task that is currently running or waiting to be executed.

- Process state – running, waiting, etc
- Program counter – location of instruction to next execute
- CPU registers – contents of all process-centric registers
- CPU scheduling information- priorities, scheduling queue pointers
- Memory-management information – memory allocated to the process
- Accounting information – CPU used, clock time elapsed since start, time limits
- I/O status information – I/O devices allocated to process, list of open files

<mark>Clustered Systems</mark>: Like multiprocessor systems, but multiple systems working together.

One of the downsides of multiprogramming is <mark>Scalability</mark>. <mark>False</mark>.

A <mark>trap</mark> or <mark>exception</mark> is a software-generated interrupt caused either by an error or a user request.

Device controller informs CPU that it has finished its operation by causing an <mark>interrupt</mark>.

There is more than one program counter in a single threaded. <mark>False</mark>.

There should be a part or all the intersection inside the memory to operate correctly. <mark>True</mark>

<mark>middleware</mark> – a set of software frameworks that provide addition services to application developers such as databases, multimedia, graphics

System programs provide a convenient environment for program development and execution. True.

Microkernel structures the operating system by removing all nonessential components from the kernel and moves as much from the kernel into user space

Communications may be via shared memory or through message passing.

System Calls are mostly accessed by programs via a high-level Application Programming Interface (API) rather than direct system call use. True.

System Calls are typically written in a high-level language (C or C++). True.

If memory location known a priori, absolute code can be generated; must recompile code if starting location changes

Threads usually share data sections. True

Threads usually share stacks. False

Data parallelism – distributes subsets of the same data across multiple cores, same operation on each.

What is the critical section for both the producer and the consumer problem? Counter.

In Amdahl's law when its 100% serial, the speedup = .. ? 1.

Who controls users thread in users space? Thread library.

One of the down sides of SJF is… ? it can cause starvation for long processes .

Peterson's Solution does not use hardware support. False.

Process synchronization can be done in hardware and software level. <mark>True</mark>.

Peterson's Solution does not meet Bounded-waiting requirement. <mark>False</mark>.

Mutex lock includes busy waiting. <mark>True</mark>.

Semaphore counting only takes 0 and 1. <mark>False</mark>.

aspen lock is generally not good in a single processor. <mark>False</mark>.

<mark>wakeup</mark> – remove one of processes in the waiting queue and place it in the ready queue.

privilege instructions only done in kernel ? <mark>true</mark>.

<mark>External Fragmentation</mark> – total memory space exists to satisfy a request, but it is not contiguous

<mark>ARM Architecture</mark> - Dominant mobile platform chip (Apple iOS and Google Android devices for example)