What is process management ?

Process management refers to the set of activities involved in creating, scheduling, and terminating processes.

A process management OS can be thought of as an instance of a program that is currently executing on a computer system

=========================================================================

As we know from the Operating Systems Evolution tree , Almost all of the OS’s existed now is based on UNIX in 1965 ,At that time there were only 2 processes, Passing by all the editions to the PDP up to **PDP-11**,all these systems were only allowed one process at a time in memory.

When a PDP-11 with memory management was obtained **(KS-11)** , The system was changed to execute several processes to remain in memory together to reduce swapping.

But this change did not apply to multiprogramming because disk I/O was synchronous.

===========================================================================

PDP-11 (Programmed Data Processor) : a series of 16-bit minicomputers sold from 1970 into the late 1990s, one of a set of products in the Programmed Data Processor (PDP) series.



===========================================================================

**Multiprogramming** : **Multi** means more than one and **Programming** means the execution of the program. when more than one program can execute in an operating system then this is termed a multiprogramming operating system.

Multiprogramming

Multiuser

allows many users to share processing time on a powerful central computer from different terminals. The operating system accomplishes this by rapidly switching between terminals

Multitasking

enables the execution of two or more programs at the same time,When a program is switched out of memory, it is temporarily saved on disk until it is required again

===========================================================================

FreeBSD can make it seem like many programs are running at the same time, and it does this by switching between different tasks (Context switching). This switching happens in these situations:

1. When a more important task becomes ready to run.

2. When there's an interruption, like handling an external event.

3. When the computer shifts between regular user work and system-level work (though not always).

4. When the computer uses a method that can stop a task and switch to another one, which is called preemptive scheduling.

===========================================================================

Scheduling

**CPU Scheduling** :It is the procedure that involves choosing which process will use the CPU to execute while another is put on hold.

**Process Scheduling**: This is the procedure by which the process management chooses a different process based on a predetermined strategy and removes an active process from the CPU.

In FreeBSD’s default scheduler, which referred to as timeshare scheduler, a process’s priority is continuously recalculated based on various parameters, such as the amount of CPU time it has used, the amount of memory resources it holds or requires for execution.

===========================================================================

Scheduling

The FreeBSD kernel implements real-time scheduling using a separate queue from the queue used for regular timeshared processes.

A process with a real-time priority is **not subject to priority degradation.**

(The priority of a real-time process won't be reduced or lowered due to factors like how long it has been running or other competing processes)

and **will only be preempted by another thread of equal or higher real-time priority.**

Real-time tasks can only be interrupted by tasks of the same or higher importance. Lower-priority tasks can't get in the way.

===========================================================================