**MULTITHREADING VS MULTITASKING**

People often get confused with these two terms, they usually consider them similar but at many levels they have differences in them. Multitasking refers to the logical extension of multiprogramming while multithreading is a thread-based multitasking. Multitasking allows CPU to execute different tasks in the same time frame, while on the other hand multithreading allows CPU to execute multiple threads of the same process simultaneously. CPU switches between programs frequently in multitasking while it switches to different threads in multithreading.

In multitasking, allocation of the memory is done for each program that CPU is executing whereas in multithreading the system allocates memory and resources to a process and the different threads of the same process use that memory and resources.

In multitasking OS, several users can be accommodated at the same time and each user have at least one program in the memory for execution.

Multithreading can share code and data and resources between multiple threads of the same process at same time. Multithreads of a single process are active in the same address space. Multithreading increases the responsiveness as if one thread is not working, the other is in working.

The advantage of multithreading over multitasking is that it is not costly as creating different task and allocating separate memory to each program and task in multitasking is quite burdensome as well as costly. Creating threads and switching between them is quite easy.