

No	Characteristic	Multiprogramming	Multiprocessing	Multithreading	Multitasking
1	What it is:	The concurrent residency of more than one program in the <b>main memory</b> is called as multiprogramming.	The availability of more than one processor per system, which can execute several set of instructions in parallel is called as multiprocessing.	A process is divided into several different sub-processes called as threads, which has its own path of execution. This concept is called as multithreading.	The execution of more than one task simultaneously is called as multitasking.
2	Number of CPU:	One	More than one	Can be one or more than one	One
3	Job processing time:	More time is taken to process the jobs.	Less time is taken for job processing.	Moderate amount of time is taken for job processing.	Moderate amount of time.
4	Number of process being executed:	One process is executed at a time.	More than one process can be executed at a time	Various components of the same process are being executed at a time.	One by one job is being executed at a time.
5	Economical:	It is economical.	Is less economical.	Is economical.	It is economical.
6	Number of users:	One at a time.	Can be one or more than one.	Usually one.	More than one.
7	Throughput:	Throughput is less.	Throughput is maximum.	Moderate.	Throughput is moderate.
8	Efficiency:	Less	Maximum	Moderate	Moderate

9	Categories:	No further divisions	Symmetric & Asymmetric.	No further divisions.	Single User & Multiuser.
10	Advantages	<p>CPU never becomes idle.</p> <p>Efficient resources utilization.</p> <p>Response time is shorter.</p> <p>Short time jobs completed faster than long time jobs.</p> <p>Increased Throughput.</p>		<p>Enhanced performance by decreased development time.</p> <p>Simplified and streamlined program coding.</p> <p>Improved GUI responsiveness.</p> <p>Simultaneous and parallelized occurrence of tasks.</p> <p>Better use of cache storage by utilization of resources.</p>	<p>Better time-management. ...</p> <p>Higher productivity. ...</p> <p>Better reaction to complex tasks. ...</p> <p>Increased responsibilities. ...</p> <p>Lower efficiency. ...</p> <p>Memory can become compromised. ...</p> <p>Greater risk of errors. ...</p> <p>Heightened stress levels.</p>
		<p><b>Increased CPU Utilization</b></p> <p><b>Increased Throughput</b></p> <p><b>Shorter Turn around Time</b></p> <p><b>Improved Memory Utilization</b></p> <p><b>Increased Resources Utilization</b></p>	<p><b>Increased Throughput</b></p> <p><b>Cost_Saving</b></p> <p><b>Increased Reliability</b></p>	<p><b>Resource Sharing</b></p> <p><b>Responsiveness</b></p> <p><b>utilization of Multiprocessor Architecture</b></p> <p><b>Economy</b></p>	<p><b><u>Timesharing:</u></b></p> <p><b><u>Handle multiple users:</u></b></p> <p><b><u>Protected memory:</u></b></p> <p><b><u>Efficient virtual memory:</u></b></p> <p><b><u>Programs can run in the background:</u></b></p> <p><b><u>Increase reliability:</u></b></p>

		<b>Multiple Users</b>			<b><u>The user can use multiple programs:</u></b>
--	--	-----------------------	--	--	---