**Multipocessor and Multicore Scheduling**

* **Granularity=** consist of the breaking down of large tasks into smaller ones. It has 5 categories:

1. **Independent parallelism=** multiple unrelated processes. There is no synchronization among processes.
2. **Coarse parallelism =** multiprocessing of concurrent processes in a multiprogramming environment.
3. **Very coarse parallelism=** distributed processing across network nodes to form a single computing environment.
4. **Medium grained parallelism =**consist with a multitasking process with a single application. Be cause of the various of thread of an application interact so frequently, scheduling decision may affect the performance of the entire application.
5. **Fine grained parallelism=** inherent in a single instruction stream, is much more complex.

* **Desing Issues =** It has 3 main issues:

1. The assignment of processes to processors
2. The use of multiprogramming on individual processors
3. The actual dispatching of a process

* **Process Scheduling =** There is a single queue for all processors.
* **Thread Scheduling =**It has 4 general approaches stand out:

1. **Load Sharing =** Is used to distinguish this strategy from load-balancing schemes in which work is allocated on a more permanent basis.

**It has several advantages:**

* The load is distributed evenly across the processors, assuring that no processor is idle while work is available to do.
* No centralized scheduler is required; when a processor is available, the scheduling routine of the operating system is run on that processor to select the next thread.

1. **Gang Scheduling =** A set of related threads is scheduled to run on a set of processors at the same time, on a one-to-one basis.
2. **Dedicated processor Assignment** = Is opposite of the Load-sharing. Each program, for the duration of its execution, is allocated a number of processors equal to the number of threads in the program. When the program terminates, the processors return to the general pool for possible allocation to another program.
3. **Dynamic Scheduling =** The number of threads in a process can be altered during the course of execution.

* **Multicore Thread Scheduling =** The most widely used contemporary OS (Windows and Linux).