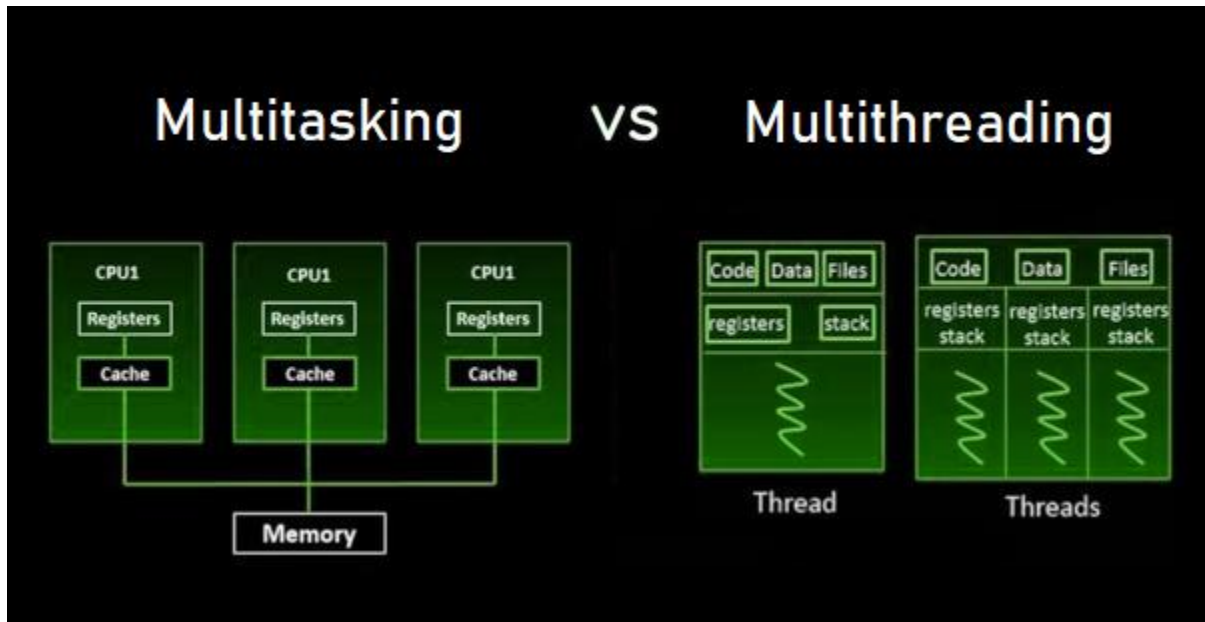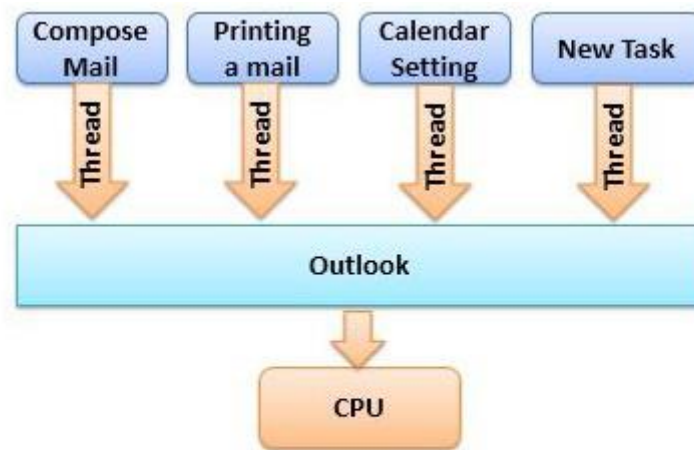# Multithreading VS Multitasking

Multithreading

In Multithreading multiple threads of the same process executed by the CPU at the same time. Or it can be defined as, Multithreading allows several threads of a single task to execute simultaneously.

To understand Multithreading, it is necessary to know about Threads.

A Thread is an elementary execution unit that contains its own set of registers, program counter and stack and shares the data, code and files to the related process.

A process can have multiple threads at once, and CPU switches between the threads so frequently that a user realize as all the threads are running recurrently. So, this is called Multithreading.

Multithreading

In multithreading system allocate some memory to a process, multiple threads of that process share the same memory and resources assigned to the process. It increases the responsiveness of system and if any one thread of the program or the process not respond, then the other thread would respond, so that the system would not be idle. Multithreading allows to share resources between threads as all threads belong to the same process can share data and code of the process.

Multitasking

In Multitasking CPU performs multiple tasks such as threads, programs or processes simultaneously. Or we can say that, when CPU execute multiple task at the same time, then it is known as Multitasking.

In multitasking the CPU switches between the tasks very rapidly, so that a user can interact and access each program instantaneously. Also, in the operating system that have an ability of multitasking, multiple users can share the system.
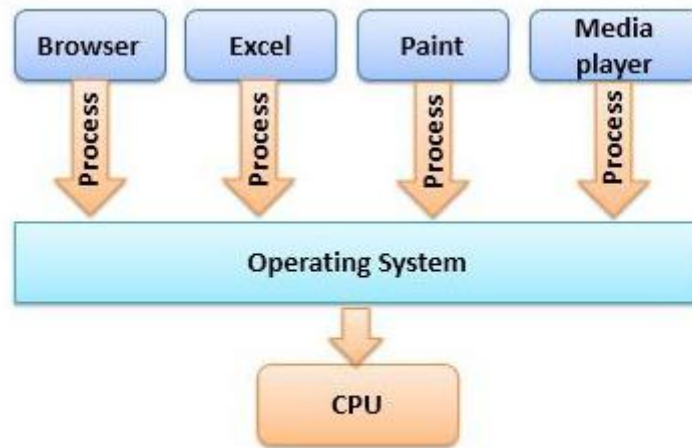
Multitasking

As it is shown that CPU frequently switches among the tasks, so it takes little time to switch from one user to another, and user realized that the entire system is running very smoothly.

In multitasking, CPU schedules each process and make at least one program in the memory for execution.

Comparison

Multithreading	Multitasking
CPU executes several threads of the process at the same time	CPU executes several tasks at the same time
CPU switches among each thread very frequently.	CPU switches among each program very frequently
System allocate same memory for the processes	System allocate separate memory for the processes
Each process is share same resources	Each process is allocated by separate resources

Key Differences Between Multithreading and Multitasking

- The basic difference between multithreading and multitasking is that, in multithreading, the system executes multiple threads of the same or different processes at the same time, but on the other hand, in multitasking, the system executes multiple programs and tasks at the same time.
- In multithreading CPU switches among multiple threads to make it look like that all threads are running simultaneously. Whereas, in multitasking, CPU switches among multiple programs so that it seems that multiple programs are running simultaneously.
- In multithreading threads belonging to the same process shares the same memory and resources as that of the process while, multitasking allocates separate memory and resources for each process/program.
- In multithreading creating threads is easy as it does not require allocating separate memory and resources for threads of the same process, but in multitasking creating a different process is costlier as the system has to allocate different memory and resources to each process.

Conclusion

Multithreading is thread-based process, in which multiple threads of a process executes at the same time active in same address space. Multitasking is similar to multiprogramming, where it performs multiple tasks and user can share system simultaneously. Multithreading is inexpensive than multitasking because threads are created easily than processes.