Multiprogramming

Definition: Multiprogramming OS is an ability of an operating system that executes more than one

program using a single processor machine. More than one task or program or jobs are present inside

the main memory at one point of time.

Buffering and spooling can overlap I/O and CPU tasks to improve the system performance but it has

some limitations that a single user cannot always keep CPU or I/O busy all the time. To increase

resource utilization, multiprogramming approaches. The OS could pick and start the execution of

one of the jobs in memory, whenever the jobs does not need CPU that means the job is working with

I/O at that time the CPU is idle at that time the OS switches to another job in memory and CPU

executes a portion of it till the job issues a request for I/O and so on.

Let’s P1 and P2 are two programs present in the main memory. The OS picks one program and

starts executing it. During execution if the P1 program requires I/O operation, then the OS will

simply switch over to P2 program. If the p2 program requires I/O then again it switches to P3 and so

on. If there is no other program remaining after P3 then the CPU will pass its control back to the

previous program.

Advantages

The advantages of multiprogramming operating system are as follows −

 CPU utilization is high because the CPU is never goes to idle state.

 Memory utilization is efficient.

 CPU throughput is high and also supports multiple interactive user terminals.

Disadvantages

The disadvantages of multiprogramming operating system are as follows −

 CPU scheduling is compulsory because lots of jobs are ready to run on CPU simultaneously.

 User is not able to interact with jobs when it is executing.

 Programmers also cannot modify a program that is being executed.

Note: If several jobs are ready in main memory and if there is not enough space for all of them, then

the system has to choose them by making a decision, this process is called job scheduling. When the

OS selects a job from the group of jobs and loads that job into memory for execution, therefore it

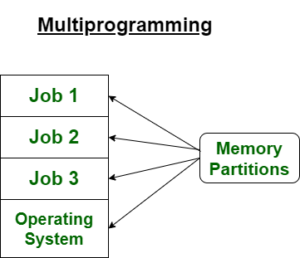
needs memory management, if several such jobs are ready then it needs CPU scheduling.

**Difference between Multiprogramming and Multitasking**

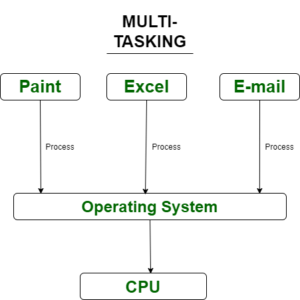
**Both Multi-programming and Multi-tasking are related to**[**Operating Systems Concepts**](https://www.geeksforgeeks.org/operating-systems/)

CPU is a super fast device and keeping it occupied for a single task is never a good idea. Considering the huge differences between CPU speed and IO speed, many concepts like multiprogramming, multitasking, multithreading, etc have been introduced to make better CPU utilisation.

**Multi programming:-**   
Multi-programming increases CPU utilisation by organising jobs (code and data) so that the CPU always has one to execute. The idea is to keep multiple jobs in main memory. If one job gets occupied with IO, CPU can be assigned to other job.



**Multi-tasking:-**  
Multi-tasking is a logical extension of multiprogramming. Multitasking is the ability of an OS to execute more than one **task** simultaneously on *a CPU* *machine.*These multiple tasks share common resources (like CPU and memory). In multi-tasking systems, the CPU executes multiple jobs by switching among them typically using a small time quantum, and the switches occur so quickly that the users feel like interact with each executing task at the same time.



**Note:**1. A task in a multitasking system is not whole application program but it can refers to a “thread of execution” when one process is divided into sub-tasks.

2. Multitasking OS are time sharing systems.

**Difference between Multiprogramming and Multi-tasking**

| **Sr.no** | **Multiprogramming** | **Multi-tasking** |
| --- | --- | --- |
| 1. | It includes the single CPU to execute the program. | It uses multiple tasks for the task allocation. |
| 2. | Concept of Context Switching is used. | Concept of Context Switching and Time Sharing is used. |
| 3. | In multiprogrammed system, the operating system simply switches to, and executes, another job when current job needs to wait. | The processor is typically used in time sharing mode. Switching happens when either allowed time expires or where there other reason for current process needs to wait (example process needs to do IO). |
| 4. | Multi-programming increases CPU utilization by organizing jobs . | In multi-tasking also increases CPU utilization, it also increases responsiveness. |
| 5. | The idea is to reduce the CPU idle time for as long as possible. | The idea is to further extend the CPU Utilization concept by increasing responsiveness Time Sharing. |
| 6. | It uses job scheduling algorithms so that more than one program can run at the same time. | Time sharing mechanism is used so that multiple tasks can run at the same time. |
| 7. | In community edition, personalized shopping experiences is not created. | Promotions, personalized shopping experiences can be displayed in enterprise edition products. |
| 8. | Execution of process takes more time. | Execution of process takes less time. |

Most of the modern operating systems use Multitasking with more complex setup.