|  |  |
| --- | --- |
| **Contiguous Memory Allocation** | **Non-Contiguous Memory Allocation** |
| It allocates only a single memory contiguous block to the process. | It separates the process into numerous blocks, each of which is assigned to a different memory address space. |
| It is very faster in execution in comparison to non-contiguous memory allocation. | It is slower in execution in comparison to contiguous memory allocation. |
| There is no overhead of address translation during execution because the process is stored in contiguous memory space in contiguous memory allocation. | There is an overhead of address translation during process execution because the process blocks are scattered across the memory space. |
| In most cases, the operating system keeps a table that lists all available and occupied partitions in the contiguous memory allocation. | In the non-contiguous memory allocation, each process must keep a table that primarily contains each block's base addresses acquired by the memory. |
| The Operating System can better control contiguous memory allocation | The Non-Contiguous Memory Allocation is difficult for the Operating System to manage. |
| Contiguous memory allocation contains two memory allocations: single partition and multi-partition. | It contains Paging and Segmentation. |
| The memory space is partitioned into fixed-sized partitions in the contiguous memory allocation, and each partition is only assigned to one process. | It is broken into several blocks, which are then placed in different areas of the memory based on available memory space. |
| Wastage of memory | No wastage of memory |
| Swapped-in processes are placed in the initially allotted space in the contiguous memory allocation. | Swapped-in processes in non-contiguous memory allocation can be organized in any location in memory. |
| Both internal and exterior fragmentation occurs. | The non-Contiguous memory allocation method causes external fragmentation. |
|  |  |

| **Parameter** | **Fixed partitioning** | **Variable partitioning** |
| --- | --- | --- |
| **Division** | Main memory is divided into fixed sized partitions. | Main memory is not divided into fixed sized partitions. |
| **Implementation** | It is easier to implement. | It is less easy to implement. |
| **Process** | Only one process can be placed in a partition. | In variable partitioning, the process is allocated a chunk of free memory. |
| **Utilize** | It does not utilize the main memory effectively. | It utilizes the main memory effectively. |
| **Fragmentation** | There is presence of internal fragmentation and external fragmentation. | There is external fragmentation. |
| **Degree** | Degree of multi-programming is less. | Degree of multi-programming is higher. |
| **Limitation** | There is limitation on size of process. | There is no limitation on size of process. |