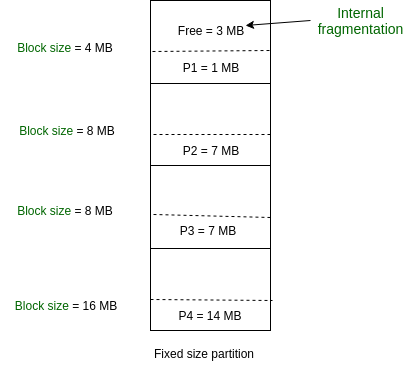
In operating systems, Memory Management is the function responsible for allocating and managing a computer’s main memory. Memory Management function keeps track of the status of each memory location, either allocated or free to ensure effective and efficient use of Primary Memory.

There are two Memory Management Techniques: **Contiguous**, and **Non-Contiguous**. In Contiguous Technique, executing process must be loaded entirely in the main memory. Contiguous Technique can be divided into:

1. Fixed (or static) partitioning
2. Variable (or dynamic) partitioning

**Fixed Partitioning:**   
This is the oldest and simplest technique used to put more than one process in the main memory. In this partitioning, the number of partitions (non-overlapping) in RAM is **fixed but the size** of each partition may or **may not be the same**. As it is a **contiguous** allocation, hence no spanning is allowed. Here partitions are made before execution or during system configure.



As illustrated in above figure, first process is only consuming 1MB out of 4MB in the main memory.   
Hence, Internal Fragmentation in first block is (4-1) = 3MB.   
Sum of Internal Fragmentation in every block = (4-1)+(8-7)+(8-7)+(16-14)= 3+1+1+2 = 7MB.

Suppose process P5 of size 7MB comes. But this process cannot be accommodated in spite of available free space because of contiguous allocation (as spanning is not allowed). Hence, 7MB becomes part of External Fragmentation.

There are some advantages and disadvantages of fixed partitioning.

**Advantages of Fixed Partitioning –**

1. **Easy to implement:**   
   Algorithms needed to implement Fixed Partitioning are easy to implement. It simply requires putting a process into a certain partition without focusing on the emergence of Internal and External Fragmentation.
2. **Little OS overhead:**   
   Processing of Fixed Partitioning requires lesser excess and indirect computational power.

**Disadvantages of Fixed Partitioning –**

1. **Internal Fragmentation:**   
   Main memory use is inefficient. Any program, no matter how small, occupies an entire partition. This can cause internal fragmentation.
2. **External Fragmentation:**   
   The total unused space (as stated above) of various partitions cannot be used to load the processes even though there is space available but not in the contiguous form (as spanning is not allowed).
3. **Limit process size:**   
   Process of size greater than the size of the partition in Main Memory cannot be accommodated. The partition size cannot be varied according to the size of the incoming process size. Hence, the process size of 32MB in the above-stated example is invalid.
4. **Limitation on Degree of Multiprogramming:**   
   Partitions in Main Memory are made before execution or during system configure. Main Memory is divided into a fixed number of partitions. Suppose if there are n1 partitions in RAM and n2 are the number of processes, then n2<=n1 condition must be fulfilled. Number of processes greater than the number of partitions in RAM is invalid in Fixed Partitioning.