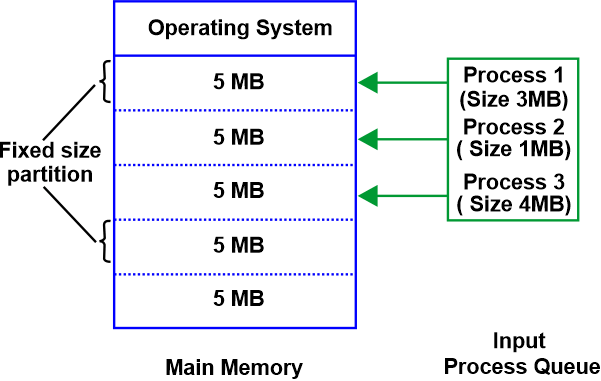
* **Fix-size Partitioning Method**
* **Flexible Partitioning Method**

Let's examine both of these strategies in depth, along with their benefits and drawbacks.

Fix-size Partitioning Method

Each process in this method of contiguous memory allocation is given a fixed size continuous block in the main memory. This means that the entire memory will be partitioned into continuous blocks of fixed size, and each time a process enters the system, it will be given one of the available blocks. Because each process receives a block of memory space that is the same size, regardless of the size of the process. Static partitioning is another name for this approach.



Three processes in the input queue in the figure above require memory space allocation. The memory has fixed-sized chunks because we are using the fixed size partition technique. In addition to the 4MB process, the first process, which is 3MB in size, is given a 5MB block. The second process, which is 1MB in size, is also given a 5MB block. So, it doesn't matter how big the process is. The same fixed-size memory block is assigned to each.

It is obvious that under this system, the number of continuous blocks into which the memory will be partitioned will be determined by the amount of space each block covers, and this, in turn, will determine how many processes can remain in the main memory at once.

The degree of multiprogramming refers to the number of processes that can run concurrently in memory. Therefore, the number of blocks formed in the RAM determines the system's level of multiprogramming.

**Advantages**

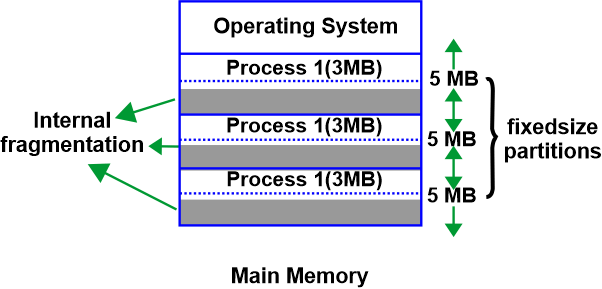
A fixed-size partition system has the following benefits:

* This strategy is easy to employ because each block is the same size. Now all that is left to do is allocate processes to the fixed memory blocks that have been divided up.
* It is simple to keep track of how many memory blocks are still available, which determines how many further processes can be allocated memory.
* This approach can be used in a system that requires multiprogramming since numerous processes can be maintained in memory at once.

**Disadvantages**

Although the fixed-size partitioning strategy offers numerous benefits, there are a few drawbacks as well:

* We won't be able to allocate space to a process whose size exceeds the block since the size of the blocks is fixed.
* The amount of multiprogramming is determined by block size, and only as many processes can run simultaneously in memory as there are available blocks.
* We must assign the process to this block if the block's size is more than that of the process; nevertheless, this will leave a lot of free space in the block. This open area might have been used to facilitate another procedure.



Flexible Partitioning Method

No fixed blocks or memory partitions are created while using this style of contiguous memory allocation technique. Instead, according on its needs, each process is given a variable-sized block. This indicates that if space is available, this amount of RAM is allocated to a new process whenever it requests it. As a result, each block's size is determined by the needs and specifications of the process that uses it.

There are no partitions with set sizes in the aforementioned diagram. Instead, the first process is given just 3MB of RAM because it requires that much. The remaining 3 processes are similarly given only the amount of space that is necessary for them.

This method is also known as dynamic partitioning because the blocks' sizes are flexible and determined as new processes start.

**Advantages**

A variable-size partitioning system has the following benefits:

* There is no internal fragmentation because the processes are given blocks of space according to their needs. Therefore, this technique does not waste RAM.
* How many processes are in the memory at once and how much space they take up will determine how many processes can be running simultaneously. As a result, it will vary depending on the situation and be dynamic.
* Even a large process can be given space because there are no blocks with set sizes.

**Disadvantages**

Despite the variable-size partition scheme's many benefits, there are a few drawbacks as well:

* This method is dynamic, hence it is challenging to implement a variable-size partition scheme.
* It is challenging to maintain record of processes and available memory space.

### Techniques for Contiguous Memory Allocation Input Queues

So far, we've examined two different contiguous memory allocation strategies. But what transpires when a fresh process needs to be given a location in the main memory? How is the block or segment that it will receive chosen?

Continuous blocks of memory assigned to processes cause the main memory to always be full. A procedure, however, leaves behind an empty block termed as a hole after it is finished. A new procedure could potentially be implemented in this area. As a result, there are processes and holes in the main memory, and each one of these holes might be assigned to a new process that comes in.

**First-Fit**

This is a fairly straightforward technique where we start at the beginning and assign the first hole, which is large enough to meet the needs of the process. The first-fit technique can also be applied so that we can pick up where we left off in our previous search for the first-fit hole.

**Best-Fit**

The goal of this greedy method, which allocates the smallest hole that meets the needs of the process, is to minimise any memory that would otherwise be lost due to internal fragmentation in the event of static partitioning. Therefore, in order to select the greatest match for the procedure without wasting memory, we must first sort the holes according to their diameters.

**Worst-Fit**

The Best-Fit strategy is in opposition to this one. The largest hole is chosen to be assigned to the incoming process once the holes are sorted based on size. The theory behind this allocation is that because the process is given a sizable hole, it will have a lot of internal fragmentation left over. As a result, a hole will be left behind that can house a few additional processes.

### Contiguous Memory Allocation's advantages and disadvantages

Contiguous memory allocation has a range of benefits and drawbacks. The following are a few benefits and drawbacks:

**Advantages**

* The number of memory blocks remaining, which affects how many further processes can be given memory space, is easy to keep track of.
* Contiguous memory allocation has good read performance since the entire file can be read from the disc in a single process.
* The contiguous allocation works well and is easy to set up.

**Disadvantages**

* Fragmentation is not an issue because each new file can be written to the disk's end after the preceding one.
* In order to choose the proper hole size while creating a new file, it needs know its final size.
* The extra space in the holes would need to be compressed or used once the diskis full.

## **Conclusion**

* When a process is brought into the main memory to be executed, contiguous memory allocation allocates contiguous blocks of memory to that process.
* There are two methods for allocating contiguous memory:
* Each process is given access to a fixed size continuous block of main memory through fixed size partitioning.
* Variable Size Partitioning: Depending on the needs of each process, space is allocated. No specific fixed-size block is present.
* There are three ways to give an entering process a hole:
* First-Fit: Assign the procedure to the first, adequate hole.
* The smallest hole that fulfils the procedure' requirements should be assigned.
* Worst-Fit: Give the entering process the hole with the biggest size among all.

