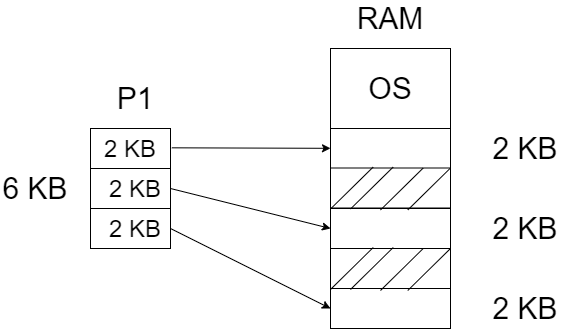
Non-Contiguous Memory Allocation | Operating System – M05 P08

This is a multipart blog article series, and in this series I am going to explain you the concepts of operating system. This article series is divided into multiple modules and this is the fifth module which consists of 26 articles.

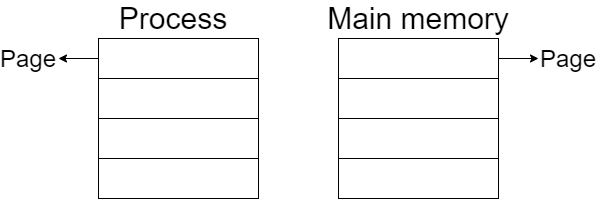
In this article we are going to see about non-contiguous memory allocation, that how it works, its pros and cons.

**Non-contiguous memory allocation**

* In contiguous memory allocation a process cannot span at different memory location, but here we can divide the process and allocate different parts of memory to it.
* External fragmentation is a problem which is present in contiguous memory allocation, can be removed by using the concept of non-contiguous memory allocation.



* Here we are dividing a process into small fragments (parts) and then allocating them to different parts of RAM, by this we are removing the problem known as external fragmentation.
* We have a problem with this approach, that the holes in the RAM are created dynamically which means their value; size and location keep on changing. So we have to analysis the RAM and divide the process, only then we can place/ put process inside memory (RAM), but this is the problem because this whole process require time and it is not time efficient.
* So to overcome this problem we divide the processes into multiple parts before it comes into the main memory (RAM), and this divided section is known as “page”.
* We are doing partition in secondary memory.
* On the other hand we also divide the main memory (RAM) and these section is known as “frame”



* Process will be divided into “pages” and main memory (RAM) will be divided into “frames”.
* Page size = Frame size
* This is because than page will fit in frame easily.

So this was all about non-contiguous memory allocation. Hope you liked it and learned something new from it.

If you have any doubt, question, quires related to this topic or just want to share something with me, than please feel free to contact me.